CONFIDENTIAL

GENESIS SOFTWARE MANUAL

New 2/6/90

SRC.

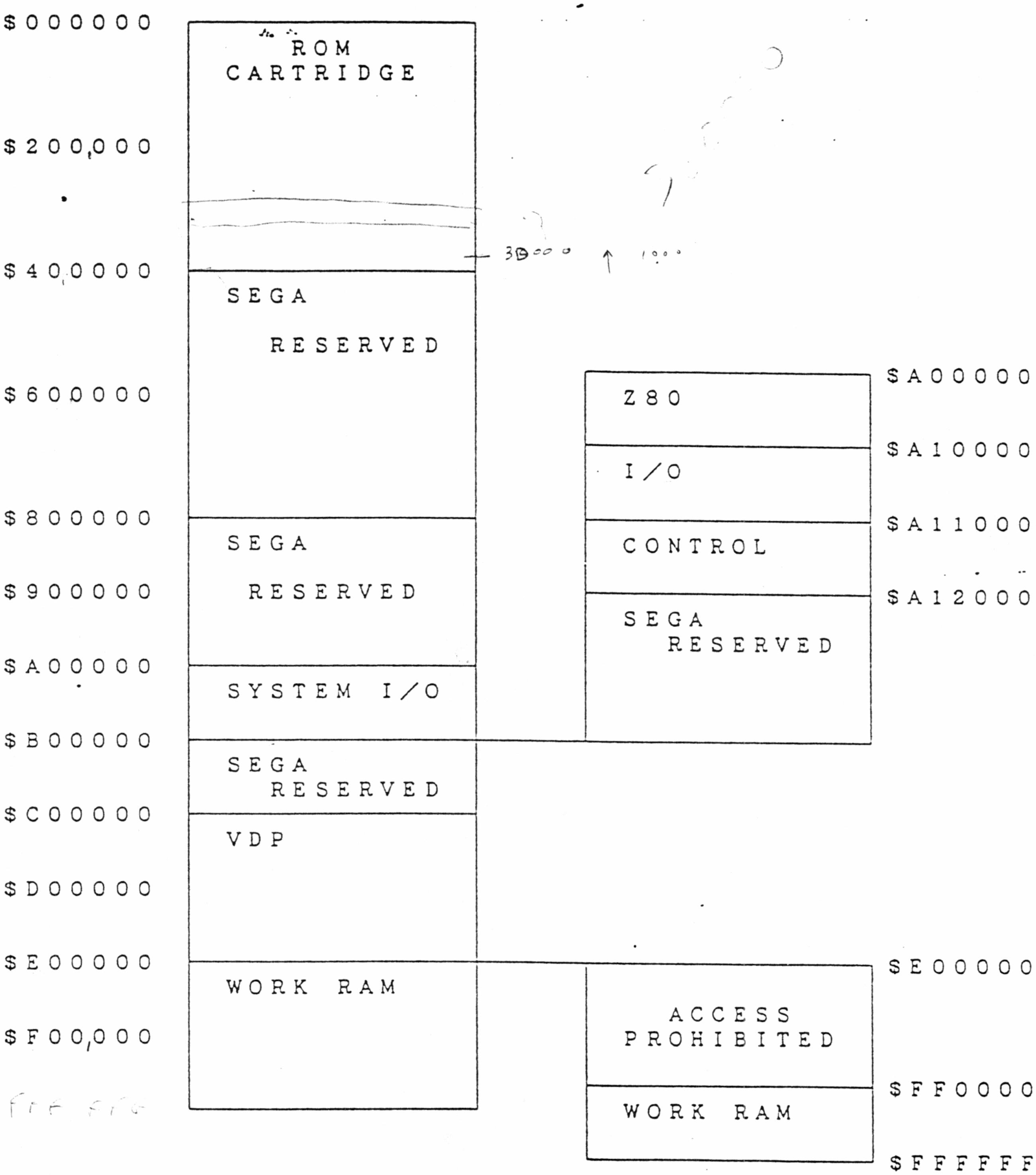
☆☆☆☆☆☆☆ INDEX ☆☆☆☆☆☆☆☆

LISA CHING's copy.

1. MEMORY MAP

§1 MEGA DRIVE 16BIT MODE (AS DISTINCT FROM MASTER SYSTEM COMPATIBILITY MODE)

◇ 68K MEMORY MAP ◇



A23 A16 A15 A14 A0

◇ Z80 MEMORY MAP ◇

Address	Area
0000H	SOUND RAM
2000H	SEGA RESERVED
4000H	SOUND CHIP (YM2612)
6000H	MISCELLANEOUS
8000H	68000BANK
FFFFH	

Area	Address
YM2612 A0	04000H
D0	04001H
A1	04002H
D1	04003H
ACCESS PROHIBITED	04004H
BANK REGISTER	06000H
ACCESS PROHIBITED	06001H
PSG 76489	07F11H
ACCESS PROHIBITED	07F12H

◇ 68000 ACCESS TO Z80 MEMORY ◇

Address	Area
$A00000	SOUND RAM
$A02000	SEGA RESERVED
$A04000	SOUND CHIP (YM2612)
$A06000	BANK REGISTER
$A08000	ACCESS PROHIBITED
$A0FFFF	

Area (upper byte)	Area (lower byte)	Address
YM2612 A0	D0	$A04000
A1	D1	$A04002
ACCESS PROHIBITED		$A04004
BANK		$A06000
ACCESS PROHIBITED		$A06002

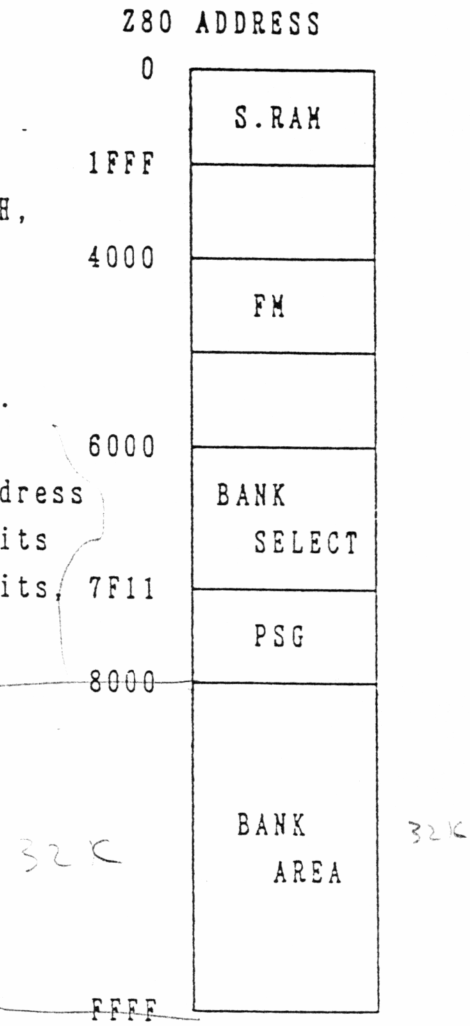
I. Z80 MAPPING

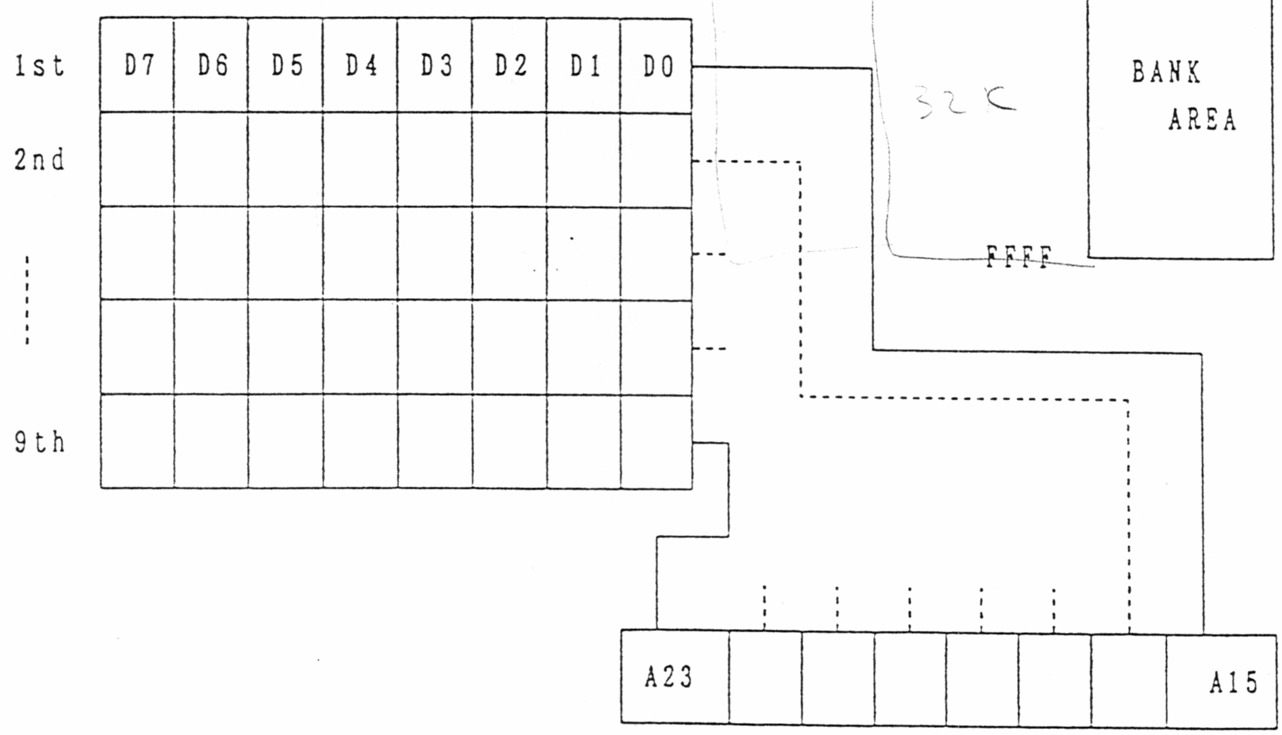
① Z80 MAP

We show the memory at right.
I/O is contained in memory map.

1) PROGRAM AREA
Prigram, data and scratch are in 0 to 1FFFH, is S-RAM.

2) BANK
From 8000H - FFFFH is window of 68K memory.
Z-80 can access all of 68K memory by BANK switching. BANK select data create 68K address from A15 to A23. You must write these 9 bits one at a time into 6000H serially, byte units, using the LSB.





◇ I/O AREA ◇

Address	Register	Function
$A10000		Version No.
$A10002	$A10003 $A10005 $A10007	DATA (CTRL 1) DATA (CTRL 2) DATA (EXP)
$A10008	$A10009 $A1000B $A1000D	CONTROL (1) CONTROL (2) CONTROL (E)
$A1000E	$A1000F $A10011 $A10013	TxDATA (1) RxDATA (1) S-MODE (1)
$A10014	$A10015 $A10017 $A10019	TxDATA (2) RxDATA (2) S-MODE (2)
$A1001A	$A1001B $A1001D $A1001F	TxDATA (E) RxDATA (E) S-MODE (E)
$A10020	ACCESS PROHIBITED	
$A1FFFF		

◇ CONTROL AREA ◇

Address	Function
$A11000	MEMORY MODE
$A11002	ACCESS PROHIBITED
$A11100	Z80 BUSREQ
$A11102	ACCESS PROHIBITED
$A11200	Z80 RESET
$A11202	ACCESS PROHIBITED
$A1FFFF	

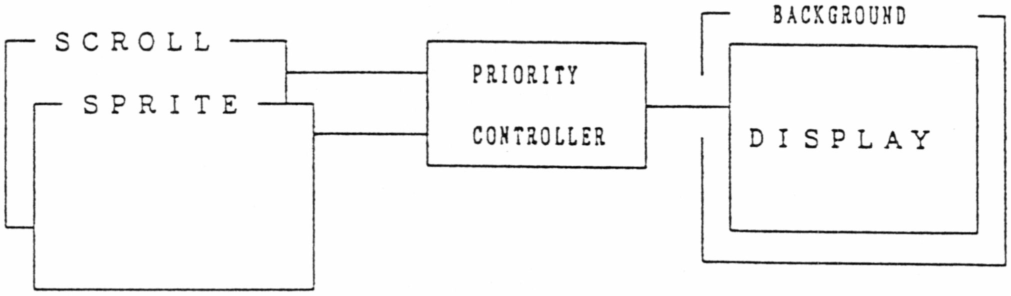
◇ VDP AREA ◇

Address	Contents	
$C00000	DATA	
$C00004	CONTROL	
$C00008	HV COUNTER	
$C0000A	ACCESS PROHIBITED	
$C00010	ACCESS PROHIBITED	PSG 76489
$C00012	ACCESS PROHIBITED	
$DFFFFF		

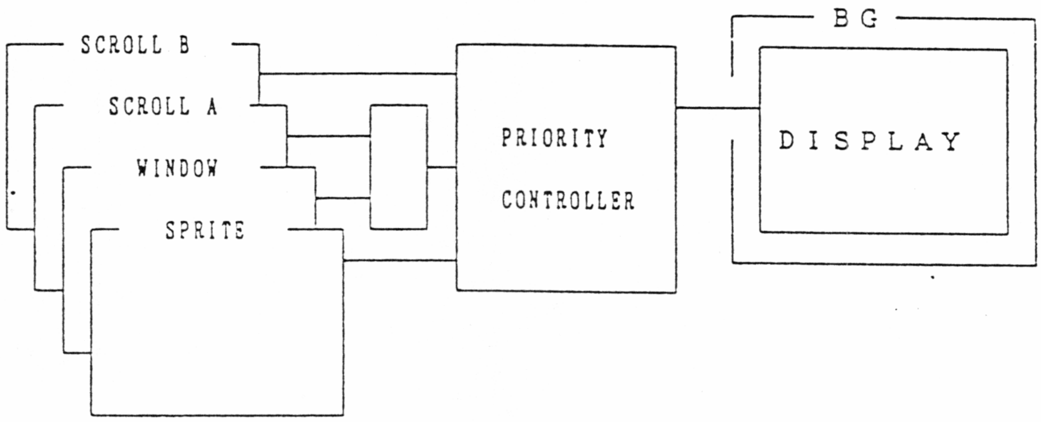
2. VDP315-5313
(Video Display Processer)

The VDP controls screen display. VDP has graphic modesⅣ and Ⅴ, where ModeⅣ is for compatibility with the MASTER SYSTEM and Ⅴ is for the new Mega drive functions. There are no advantages to using modeⅣ, so it is assumed that all Mega drive development will use modeⅤ.
In modeⅤ, the VDP display has 4 planes: SPRITE, SCROLLA/WINDOW, SCROLLB, and BACKGROUND

GRAPHICⅣ MODE (COMPATIBILITY MODE)



GRAPHICⅤ MODE (16BIT MODE)



◇ TERMINOLOGY ◇

1. A unit of position on X. Y coodinates is called a "DOT".
2. A minimum unit of display is called a "PIXEL".
3. "CELL" means an 8 (pixel) × 8 (pixel) pattern.
4. SCROLL indicated a repositionable screen-spanning play field.
5. CPU usually indicates the 68000.
6. VDP stands for Video Display Processor.
7. CTRL stands for Control.
8. VRAM stands for VDP RAM, the 64K bytes area of RAM accessible only through the VDP.
9. CRAM stands for Color RAM, 64 9 bit words inside the VDP chip.
10. VSRAM stands for vertical Scroll RAM, 40 10bit words inside the VDP chip.
11. DMA stands for Direct Memory Access, the process by which the VDP performs high speed fills or memory copies.
12. PSG stands for Programmable sound Generator, a class of low-capability Sound chips. The Mega drive contains a Texas Instruments 76489 PSG chip.
13. FM stands for Frequency Modulation, a class of high-capability sound chip. The Mega drive contains a Yamaha 2612 FM chip.

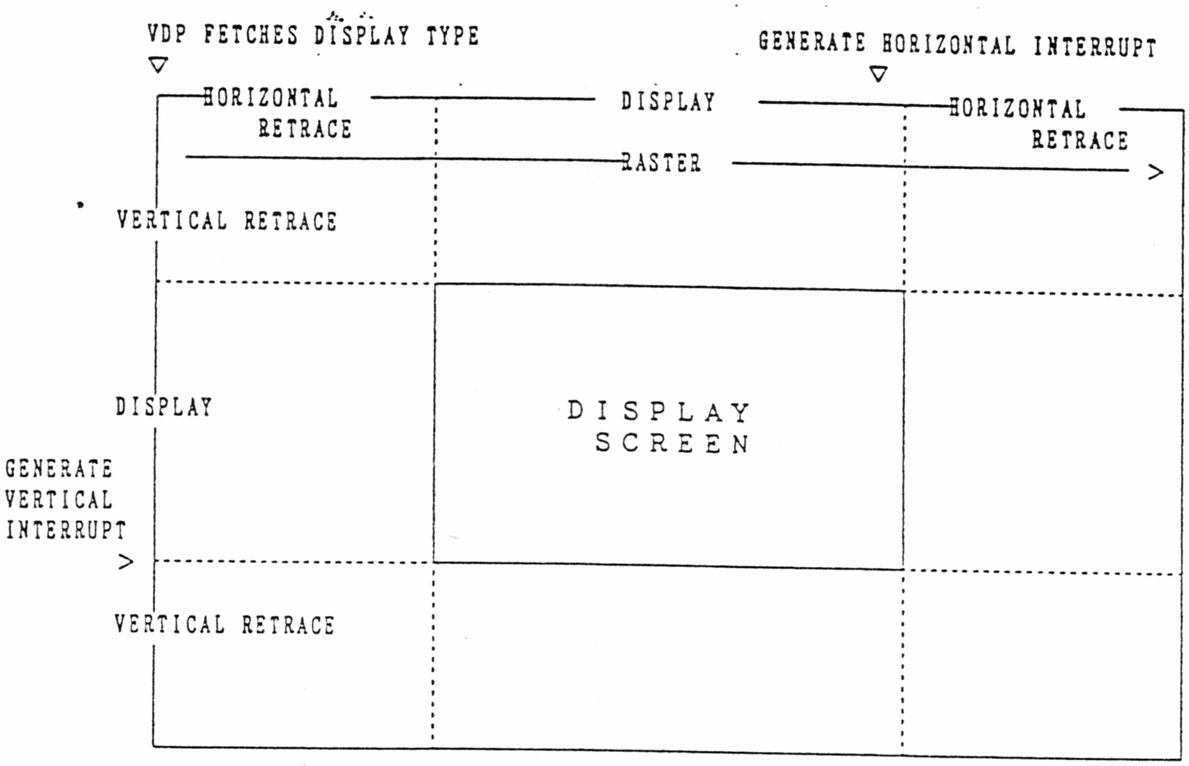
§1 DISPLAY SPECIFICATION

DISPLAY SPECIFICATION OUTLINE

DISPLAY SIZE	THERE ARE TWO MODES; 32*28CELL (256*224PIXEL) 40*28CELL (320*224PIXEL)
CHARACTER GENARATER	8*8CELLS 1300-1800 depending on general system configuration.
SCROLL PLAYFIELDS	Two scrolling play fields, whose size in cells is selectable from: 32*32, 32*64, 32*128 64*32, 64*64 128*32
SPRITE	Sprite size is programable on a sprite by sprite basis, with the following choices. 8*8, 8*16, 8*24, 8*32 16*8, 16*16, 16*24, 16*32 24*8, 24*16, 24*24, 24*32 32*8, 32*16, 32*24, 32*32 There are 64 sprites available when the screen is in 32 cell wide mode, or 80 when the screen is in 40 cell wide mode.
WINDOW	1 window associated with the Scroll A play field.
COLORS	64 colors/512 possibilities

For PAL (the European Television 50HZ standard), a vertical size of 30 cells (240 dots) is selectable.

The VDP supports both NTSC and PAL television standards. In both cases, the screen is divided into active scan, where the picture is displayed, and vertical retrace(or vertical blanking) where the monitor prepares for the next display.



Numbers of rasters in a screen are as follows:

	LINES/SCREEN	VCELL NO	LINE NO. (DISPLAY)	LINE NO. (RETRACE)
NTSC	262	28	224 RASTER	38 RASTER
PAL	312	28	224 RASTER	98 RASTER
		30	240 RASTER	82 RASTER

§2 VDP STRUCTURE

The CPU controls the VDP by special I/O memory locations.

◇ CTRL (control) ◇

This controls REGISTER, VRAM, CRAM, VSRAM, DMA DISPLAY, etc.

◇ VRAM (VDP RAM) ◇

General, purpose storage area for display data.

◇ CRAM (COLOR RAM) ◇

64 colors divided into 4 palettes of 16 colors each.

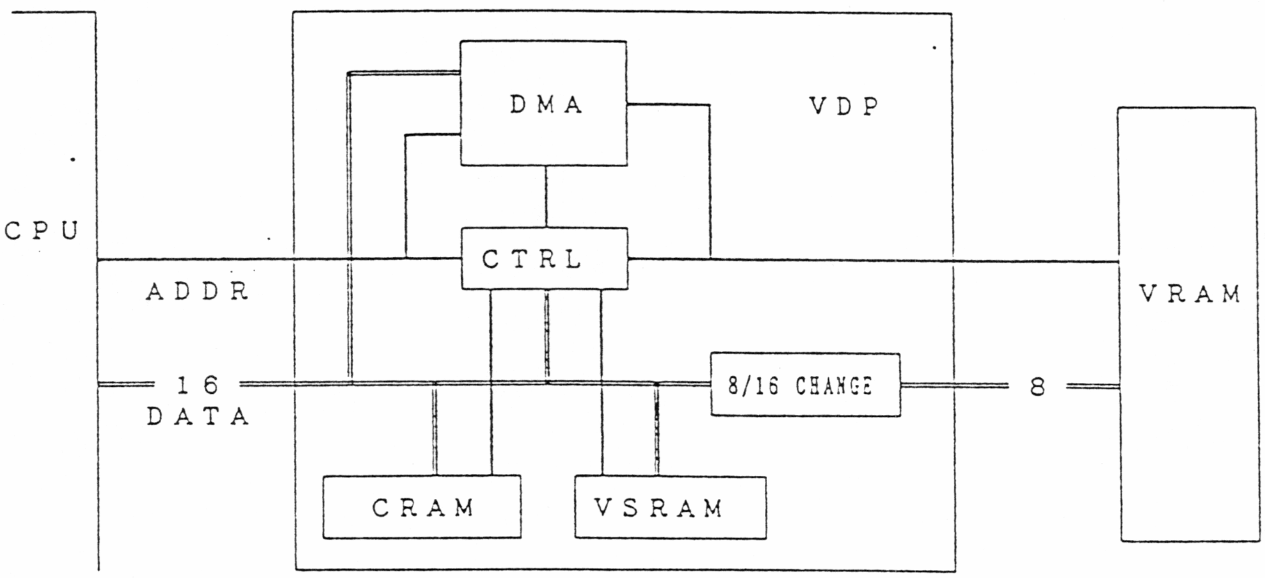
◇ VSRAM (Vertical scroll RAM) ◇

Up to 20 different vertical scroll values each for scrolling play fields A and B.

◇ DMA (Direct Memory Access) ◇

The VDP may move data at high speed from CPU memory to VRAM, CRAM, and VSRAM instead of the CPU, bytaking the 68000 off the bus and doing DMA itself.

The VDP can also fill the VRAM with a constant, or copy from VRAM to VRAM without disturbing the 68000.



§ 3 INTERRUPT

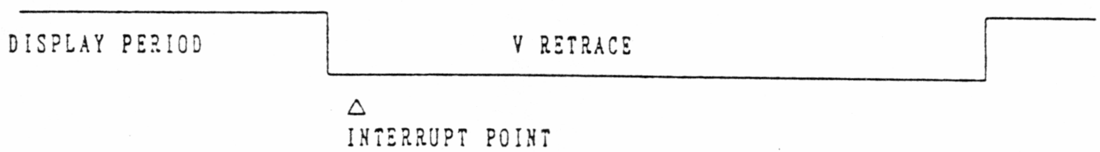
There are three interrupts, Vertical, Horizontal, and External. You can control each interrupt by the IE0, IE1, and IE2 bits in the VDP registers. The interrupts use the AUTO-VECTOR mode of the 68000 and are at levels 6, 4, and 2 respectively, the level 6 vertical interrupt having the highest priorty.

IE0	V Interrupt	(LEVEL6)
IE1	H Interrupt	(LEVEL4)
IE2	External Interrupt	(LEVEL2)

1 : Enable
0 : Disable

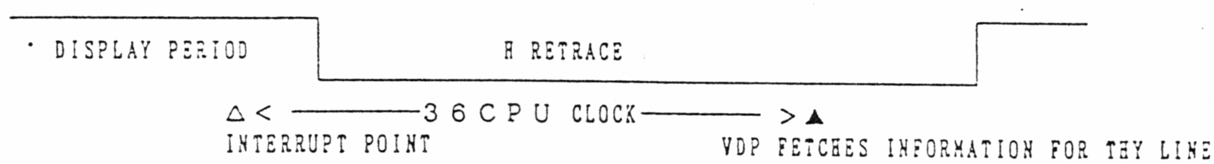
◇ VERTICAL INTERRUPT (V-INT) ◇

The vertical interrupt occurs just after V retrace.

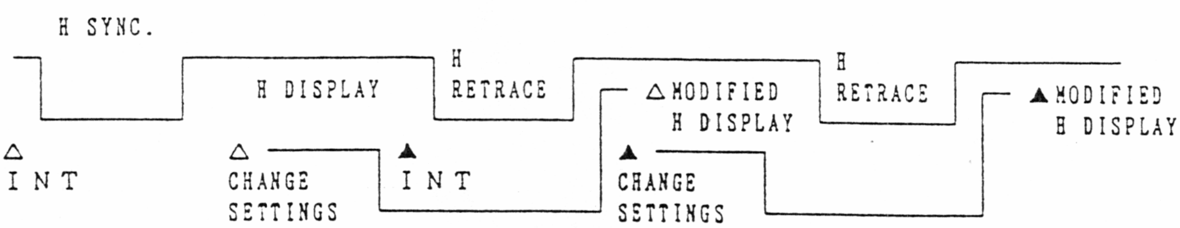


◇ HORIZONTAL INTERRUPT (H-INT) ◇

The horizontal interrupt occurs just before H retrace.



The VDP loads the required display information, including all required register values, for the line in about 36clocks, thus the CPU cancontrol the display of the next line but not the line on which the interrupt occurs.

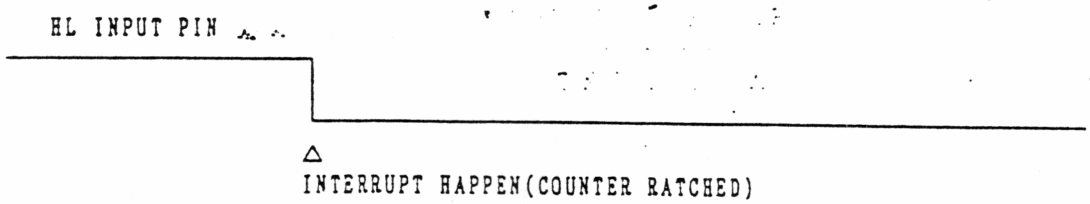


The horizontal interrupt is controlled by a line counter in register #10.
If this line counter is changed at each interrupt, the desired spacing of interrupts may be achieved.

Thus: If Register #10 equals 00h then the interrupt occurs every line.
If Register #10 equals 01h then the interrupt occurs every other line.
If Register #10 equals 02h then the interrupt occurs every third line.

◇ EXTERNAL INTERRUPT (EX-INT) ◇

The external interrupt is generated by a peripheral device (gun, modem) and stops the H, V counter for later examination by the CPU.



Please see other sections of this manual for information about the H, V counter and the initialization of the external interrupt.

§4 VDP PORT

The VDP ports are at location 68000 in the 68000 memory space.

Address	UPPER BYTE	LOWER BYTE
$C00000	DATA PORT	
$C00002	"	
$C00004	CONTROL PORT	
$C00006	"	
$C00008	HV COUNTER	
$C0000A	PROHIBITED	
$C0000C	PROHIBITED	
$C0000E	PROHIBITED	
$C00010	PROHIBITED	PSG

◇ $C00000 (DATA PORT) ◇

READ/WRITE: VRAM, VSRAM, CRAM

$C00000									
	DT15	DT14	DT13	DT12	DT11	DT10	DT9	DT8	(D15~ D8)
	DT7	DT6	DT5	DT4	DT3	DT2	DT1	DT0	(D7 ~ D0)

※ $C00000 and $C00002 are functionally equiralent.

◇ $C00004 (CONTROL PORT) ◇

READ : STATUS REGISTER

$C00004									
	*	*	*	*	*	*	EMPT	FULL	(D15~ D8)
	F	SOVR	C	ODD	VB	HB	DMA	PAL	(D7 ~ D0)

* NO USE

EMPT···· 1 : WRITE FIFO EMPTY
0 :
FULL···· 1 : WRITE FIFO FULL
0 :
F········· 1 : V interrupt happened.
SOVR···· 1 : Sprites overflow occurred,too many in one line.
Over 17 in 32 cell mode.
Over 21 in 40 cell mode.
C········· 1 : Collision happened between non-zero pixels
in two sprites.
0 :
ODD······ 1 : Odd frame in interlace mode.
0 : Even frame in interlace mode.
VB······· 1 : During V blanking
0 :
HB······· 1 : During H blanking
0 :
DMA······ 1 : DMA BUSY
0 :
PAL······ 1 : PAL MODE
0 : NTSC MODE

WRITE1 : REGISTER SET

$C00004	1	0	0	RS4	RS3	RS2	RS1	RS0	(D15~ D8)
	D7	D6	D5	D4	D3	D2	D1	D0	(D7 ~ D0)

※ $C00004 and $C00006 are functionally equivalent.

RS4~RS0 : Register No.
D7~D0 : Data

※ You must use word or long word access to the VDP ports when setting the registers.
Long word access is equivalelnet to two word accesses, with D31-D16 written first.

WRITE2 : ADDRESS SET

1st $C00004	CD1	CD0	A13	A12	A11	A10	A9	A8	(D15~ D8)
	A7	A6	A5	A4	A3	A2	A1	A0	(D7 ~ D0)

2nd $C00004	0	0	0	0	0	0	0	0	(D15~ D8)
	CD5	CD4	CD3	CD2	0	0	A15	A14	(D7 ~ D0)

CD5~CD0 : ID CODE
A15~A0 : DESTINATION RAM ADDRESS

ACCESS MODE	CD5	CD4	CD3	CD2	CD1	CD0
VRAM WRITE	0	0	0	0	0	1
CRAM WRITE	0	0	0	0	1	1
VSRAM WRITE	0	0	0	1	0	1
VRAM READ	0	0	0	0	0	0
CRAM READ	0	0	1	0	0	0
VSRAM READ	0	0	0	1	0	0

※ You must use word or long word when performing these operations.

◇ $C00008 (HV COUNTER) ◇

NON INTERLACE MODE

$C00008

VC7	VC6	VC5	VC4	VC3	VC2	VC1	VC0	(D15～ D8)
HC8	HC7	HC6	HC5	HC4	HC3	HC2	HC1	(D7 ～ D0)

INTERLACE MODE

$C00008

VC7	VC6	VC5	VC4	VC3	VC2	VC1	VC8	(D15～ D8)
HC8	HC7	HC6	HC5	HC4	HC3	HC2	HC1	(D7 ～ D0)

HC8～HC1 : H COUNTER
VC8～VC0 : V COUNTER

O PATTERN NAME TABLE BASE ADDRESS FOR SCROLL B

REG. #4

	MSB						LSB
0	0	0	0	0	SB15	SB14	SB13

VRAM ADDR %XXX0_0000_0000_0000

only on $2000 byte bountries

O SPRITE ATTRIBUTE TABLE BASE ADDRESS

REG. #5

	MSB						LSB
0	AT15	AT14	AT13	AT12	AT11	AT10	AT9

AT9 should be 0 in H40 cell mode
VRAM ADDR %XXXX_XXX0_0000_0000 (32cell) only on $200 byte
VRAM ADDR %XXXX_XX00_0000_0000 (40cell) only on $400 byte

REG. #6

	MSB						LSB
0	0	0	0	0	0	0	0

O BACKGROUND COLOR

REG. #7

	MSB						LSB
0	0	CPT1	CPT0	COL3	COL2	COL1	COL0

CPT1, 0 : COLOR PALLET
COL3~0 : COLOR CODE

REG. #8

	MSB						LSB
0	0	0	0	0	0	0	0

REG. #9

	MSB						LSB
0	0	0	0	0	0	0	0

O H INTERRUPT REGISTER

REG. #10

	MSB						LSB
HIT7	HIT6	HIT5	HIT4	HIT3	HIT2	HIT1	HIT0

This register makes H interrupt timing by number of laster.
H interrupt is enabled by IE=1

○ MODE SET REGISTER No. 3

REG. #11	MSB						LSB
	0	0	0	0	IE2	VSCR	HSCR	LSCR

IE2······ 1: Enable external interrupt (68000LEVEL2)
0: Disable external interrupt
※ See INTERRUPT and SYSTEM I/O

VSCR: V scroll mode

VSCR	FUNCTION
0	FULL SCROLL
1	EACH 2CELL SCROLL

HSCR, LSCR: H scroll mode

HSCR	LSCR	FUNCTION
0	0	FULL SCROLL
0	1	PROHIBITED
1	0	EACH 1CELL SCROLL
1	1	EACH 1LINE SCROLL

※ BOTH SCROLL A and B

○ MODE SET REGISTER No. 4

REG. #12	MSB							LSB
	RS0	0	0	0	S/TE	LSM1	LSM0	RS1

RS0······ 0: HORIZONTAL 32 CELL MODE
1: 〃 40 〃
RS1······ 0: HORIZONTAL 32 CELL MODE
1: 〃 40 〃
※ You should set same No. in RS0,RS1.
32cell 0000_XXX0
40cell 1000_XXX1
S/TE···· 1: Enable SHADOW and HILIGHT.
0: Disable 〃
LSM1, LSM0: Interlase mode setting

LSM1	LSM0	FUNCTION
0	0	NO INTERLASE
0	1	INTERLASE
1	0	PROHIBITED
1	1	INTERLASE (DOUBLE RESOLUTION)

○ H SCROLL DATA TABLE BASE ADDRESS

REG. #13	MSB							LSB
	0	0	HS15	HS14	HS13	HS12	HS11	HS10

only on $400

VRAM ADDR XXXXX_XX00_0000_0000

REG. #14	MSB							LSB
	0	0	0	0	0	0	0	0

O AUTO INCREMENT DATA

This regoster controls bias number of increment data.

MSB LSB

REG. #15								
	INC7	INC6	INC5	INC4	INC3	INC2	INC1	INC0

INC7～0: Bias number (0～$FF)
This number is added automatically after ram access.

O SCROLL SIZE

MSB LSB

REG. #16								
	0	0	VSZ1	VSZ0	0	0	HSZ1	HSZ0

VSZ1, 0: VSIZE

VSZ1	VSZ0	FUNCTION
0	0	V 32cell
0	1	V 64cell
1	0	PROHIBITED
1	1	V 128cell

HSZ1, 0: HSIZ

HSZ1	HSZ0	FUNCTION
0	0	H 32cell
0	1	H 64cell
1	0	PROHIBITED
1	1	H 128cell

※ Both of scroll A and B

O WINDOW H POSITION

MSB LSB

REG. #17								
	RIGT	0	0	WHP5	WHP4	WHP3	WHP2	WHP1

RIGT···· 0: Window is in left side from base point.
1: Window is in right side from base point.
WHP5～1 Base pointer 0=Left side
1=1cell right
2···

O WINDOW V POSITION

MSB LSB

REG. #18								
	DOWN	0	0	WVP4	WVP3	WVP2	WVP1	WVP0

DOWN···· 0: Window is in upper side from base point.
1: Window is in lower side from base point.
WVP4～0 Base pointer 0=Upper side
1=1cell down
2···

O DMA LENGTH COUNTER LOW

	MSB							LSB
REG. #19	LG7	LG6	LG5	LG4	LG3	LG2	LG1	LG0

O DMA LENGTH COUNTER HIGH

	MSB							LSB
REG. #20	LG15	LG14	LG13	LG12	LG11	LG10	LG9	LG8

LG15~0: DMA LENGTH COUNTER

O DMA SOURCE ADDRESS LOW

	MSB							LSB
REG. #21	SA8	SA7	SA6	SA5	SA4	SA3	SA2	SA1

O DMA SOURCE ADDRESS MID.

	MSB							LSB
REG. #22	SA16	SA15	SA14	SA13	SA12	SA11	SA10	SA9

O DMA SOURCE ADDRESS HIGH

	MSB							LSB
REG. #23	DMD1	DMD0	SA22	SA21	SA20	SA19	SA18	SA17

SA22~1: DMA SOURCE ADDRESS
DMD1, 0: DMA MODE

DMD1	DMD0	FUNCTION
0	SA23	MEMORY TO VRAM
1	0	VRAM FILL
1	1	VRAM COPY

§6 ACCESS VDP RAM

◇RAM ADDRESS SETING◇

You can access VRAM CRAM and VSRAM after writing 32bits of control data to $C00004 or $C00006.

You have to use word or long word when addressing. If you use long word D31~D16 is 1st, D15~D0 2nd.

1st
$C00004

CD1	CD0	A13	A12	A11	A10	A9	A8	(D15~ D8)
A7	A6	A5	A4	A3	A2	A1	A0	(D7 ~ D0)

2nd
$C00004

0	0	0	0	0	0	0	0	(D15~ D8)
CD5	CD4	CD3	CD2	0	0	A15	A14	(D7 ~ D0)

CD5~CD0 : ID CODE
A15~A0 : DESTINATION RAM ADDRESS

アクセスモード	CD5	CD4	CD3	CD2	CD1	CD0
VRAM WRITE	0	0	0	0	0	1
CRAM WRITE	0	0	0	0	1	1
VSRAM WRITE	0	0	0	1	0	1
VRAM READ	0	0	0	0	0	0
CRAM READ	0	0	1	0	0	0
VSRAM READ	0	0	0	1	0	0

◇ VRAM ACCESS ◇

VRAM address range from 0 to 0FFFFH, 64K bytes total.
VRAM access addressing is as follow when writing:

1st
$C00004

0	1	A13	A12	A11	A10	A9	A8	(D15~ D8)
A7	A6	A5	A4	A3	A2	A1	A0	(D7 ~ D0)

2nd
$C00004

0	0	0	0	0	0	0	0	(D15~ D8)
0	0	0	0	0	0	A15	A14	(D7 ~ D0)

A15~A0 : VRAM address

Data
$C00000

D15	D14	D13	D12	D11	D10	D9	D8	(D15~ D8)
D7	D6	D5	D4	D3	D2	D1	D0	(D7 ~ D0)

D15~D0 : VRAM data

When you use long word D31~D16 is 1st, D15~D0 2nd. When you do byte writing, data is D7~D0, and may be written to $C00000 or $C00001.
VRAM address is increased by the value of REGISTER #15, independent data size. VRAM address A0 is used in the calculation of the address increment, but is ignored during address decoding.
VRAM addressing and decoding are as follows; the VRAM adress decode uses A15~A1, and A0 specifies the data write format. Write data cannot cross a word boundary, high and low bytes are exchanged if A0=1.

GTB #13

A0=0

ADDRESS:	BYTE	WORD	LONG WORD
EVEN ODD	~~D7~D0~~	D15~D8 D7~D0	D31~D24 D23~D16
EVEN ODD			D15~D8 D7~D0

A0=1

ADDRESS:	BYTE	WORD	LONG WORD
EVEN ODD	~~D7~D0~~	D7~D0 D15~D8	D23~D16 D31~D24
EVEN ODD			D7~D0 D15~D8

(EXAMPLE)

START ADDRESS: 0 REG. #15=2

ADDRESS	~~BYTE~~ GTB #13	WORD	LONG WORD
0 1	~~1st D7~D0~~	1st D15~D8 D7~D0	1st D31~D24 D23~D16
2 3	~~2nd D7~D0~~	2nd D15~D8 D7~D0	1st D15~D8 D7~D0
4 5	~~3rd D7~D0~~	3rd D15~D8 D7~D0	2nd D31~D24 D23~D16
6 7	~~4th D7~D0~~	4th D15~D8 D7~D0	2nd D15~D8 D7~D0
8 9	~~5th D7~D0~~	5th D15~D8 D7~D0	3rd D31~D24 D23~D16

START ADDRESS: 0 REG. #15=1

ADDRESS	~~BYTE~~	WORD	LONG WORD
0 1	~~2nd D7~D0~~ ~~1st D7~D0~~	2nd D7~D0 D15~D8	1st D7~D0 D15~D8
2 3	~~4th D7~D0~~ ~~3rd D7~D0~~	4th D7~D0 D15~D8	2nd D7~D0 D15~D8
4 5	~~6th D7~D0~~ ~~5th D7~D0~~	6th D7~D0 D15~D8	3rd D7~D0 D15~D8
6 7	~~8th D7~D0~~ ~~7th D7~D0~~	8th D7~D0 D15~D8	4th D7~D0 D15~D8
8 9	~~10th D7~D0~~ ~~9th D7~D0~~	10th D7~D0 D15~D8	5th D7~D0 D15~D8

START ADDRESS: 1 REG. #15=2

GTB #18

ADDRESS:	BYTE	WORD	LONG WORD
0 1	~~1st D7~D0~~	1st D7~D0 D15~D8	1st D23~D16 D31~D24
2 3	~~2nd D7~D0~~	2nd D7~D0 D15~D8	1st D23~D16 D31~D24
4 5	~~3rd D7~D0~~	3rd D7~D0 D15~D8	2nd D23~D16 D31~D24
6 7	~~4th D7~D0~~	4th D7~D0 D15~D8	2nd D23~D16 D31~D24
8 9	~~5th D7~D0~~	5th D7~D0 D15~D8	3rd D23~D16 D31~D24

START ADDRESS: 1 REG. #15=1

ADDRESS:	BYTE	WORD	LONG WORD
0 1	~~1st D7~D7~~	1st D7~D0 D15~D8	1st D23~D16 D31~D24
2 3	~~3rd D7~D7~~ ~~2nd D7~D7~~	3rd D7~D0 D15~D8	2nd D23~D16 D31~D24
4 5	~~5th D7~D7~~ ~~4th D7~D7~~	5th D7~D0 D15~D8	3rd D23~D16 D31~D24
6 7	~~7th D7~D7~~ ~~6th D7~D7~~	7th D7~D0 D15~D8	4th D23~D16 D31~D24
8 9	~~9th D7~D7~~ ~~8th D7~D7~~	9th D7~D0 D15~D8	5th D23~D16 D31~D24

VRAM READ

1st
$C00004

0	0	A13	A12	A11	A10	A9	A8	(D15~ D8)
A7	A6	A5	A4	A3	A2	A1	A0	(D7 ~ D0)

2nd
$C00004

0	0	0	0	0	0	0	0	(D15~ D8)
0	0	0	0	0	0	A15	A14	(D7 ~ D0)

A15~A0 : VRAM ADDRESS

Data
$C00000

D15	D14	D13	D12	D11	D10	D9	D8	(D15~ D8)
D7	D6	D5	D4	D3	D2	D1	D0	(D7 ~ D0)

D15~D0 : VRAM DATA

The data is always read in word units. A0 is ignored during the read; no swap of bytes occurs if A0=1.

Subsegvent reads are from address incremented by REGISTER #15. A0 is used in calculation of the next address.

◇ CRAM ACCESS ◇

The CRAM contains 128 bytes, adresses 0 to 7FH.
For word wide writes to the CRAM, use:

1st $C00004	1	1	0	0	0	0	0	0	(D15~ D8)
	0	A6	A5	A4	A3	A2	A1	A0	(D7 ~ D0)

2nd $C00004	0	0	0	0	0	0	0	0	(D15~ D8)
	0	0	0	0	0	0	0	0	(D7 ~ D0)

A6~A0 : CRAM ADDRESS

DATA $C00000	0	0	0	0	B2	B1	B0	0	(D15~ D8)
	G2	G1	G0	0	R2	R1	R0	0	(D7 ~ D0)

D15~D0 are valid when we use word for data set. If the writes are byte wide, write the high byte to $C00000 and the low byte to $C00001.
A long word wide access is equivalent to two sequential word wide accesses. Place the first data in D31-D16, and the second data in D15-D0.
The data may be written sequentially; the address is incremented by the value of REGISTER #15 after every write, independent of whether the width is byte or word.
Note that A0 is used in the increment but not in address decoding, resulting in some interesting side-effects if writes are attempted at odd addresses.

For word wide reads from the CRAM, use:

1st $C00004	0	0	0	0	0	0	0	0	(D15~ D8)
	0	A6	A5	A4	A3	A2	A1	A0	(D7 ~ D0)

2nd $C00004	0	0	0	0	0	0	0	0	(D15~ D8)
	0	0	1	0	0	0	0	0	(D7 ~ D0)

A6~A0 : CRAM Address

Data $C00000	*	*	*	*	B2	B1	B0	*	(D15~ D8)
	G2	G1	G0	*	R2	R1	R0	*	(D7 ~ D0)

* 不定

◇ VSRAM ACCESS ◇

The VSRAM contains 80 bytes, addresses 0 to 4FH.
For word wide writes to the VSRAM, use:

1st
$C00004

0	1	0	0	0	0	0	0	(D15~ D8)
0	A6	A5	A4	A3	A2	A1	A0	(D7 ~ D0)

2nd
$C00004

0	0	0	0	0	0	0	0	(D15~ D8)
0	0	0	1	0	0	0	0	(D7 ~ D0)

A6~A0 : VSRAM Address

Data
$C00000

					VS10	VS9	VS8	(D15~ D8)
VS7	VS6	VS5	VS4	VS3	VS2	VS1	VS0	(D7 ~ D0)

VS10~VS0 : V quantity of scroll

If you use word for data and valid in D15-D0.
D15~D0 are valid when we use word for data set.
If the writes are byte wide, write the high byte to $C00000 and the low byte to $C00001.
A long word wide access is equivalent to two sequential word wide accesses. Place the first data in D31-D16, and the second data in D15-D0.
The data may be written sequentially; the address is incremented by the value of REGISTER #15 after every write, independent of whether the width is byte or word.
Note that A0 is used in the increment but not in address decoding, resulting in some interesting side-effects if writes are attempted at odd addresses.

For word wide reads from the VSRAM, use:

1st
$C00004

0	0	0	0	0	0	0	0	(D15~ D8)
0	A6	A5	A4	A3	A2	A1	A0	(D7 ~ D0)

2nd
$C00004

0	0	0	0	0	0	0	0	(D15~ D8)
0	0	0	1	0	0	0	0	(D7 ~ D0)

A6~A0 : VSRAM Address

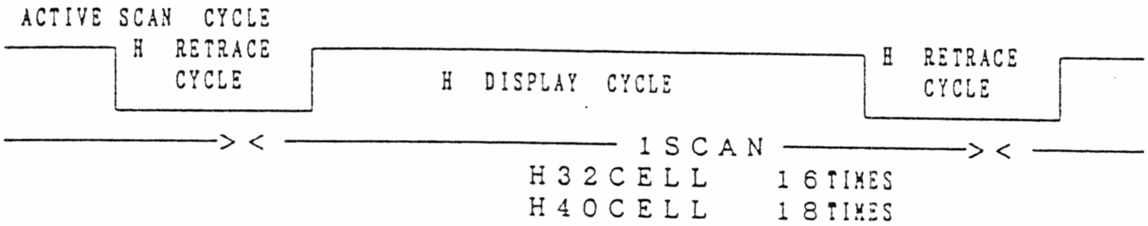
Data
$C00000

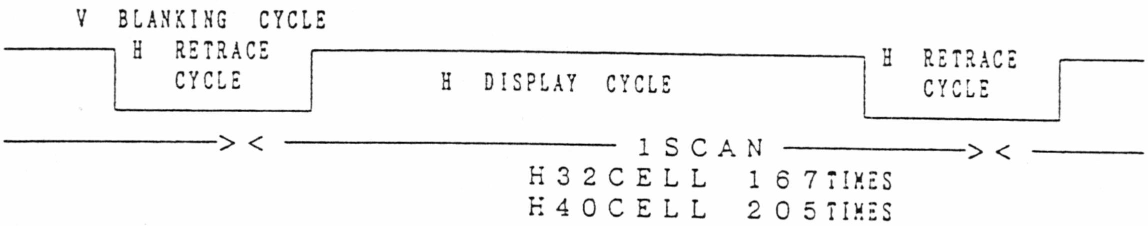
					VS10	VS9	VS8	(D15~ D8)
VS7	VS6	VS5	VS4	VS3	VS2	VS1	VS0	(D7 ~ D0)

VS10~VS0 : V quantity of scroll

◇ ACCESS TIMING ◇

The CPU and VDP access VRAM, CRAM and VSRAM using timesharing. Because the VDP is very busy during the active scan, the CPU accesses are limited. However, during vertical blanking the CPU may access the VDP continuously.
The number of permitted accesses by the CPU additionally depends on whether the screen is in 32 cell mode or 40 cell mode. Additionally, the access size depends on the RAM type; a VRAM access is byte wide, but CRAM and VSRAM are word wide.





For example, in 32 cell mode, the CPU may access the VRAM 16 times during horizontal scan in a single line. Each access is a byte write, so this amounts to 8 words. However CRAM and VSRAM, though sharing the 16 time limit, are word accesses so that 16 words may be written in a single line.
Although there is a four-word FIFO, if writes are done in a tight loop during active scan the FIFO will fill up and the CPU will eventually end up waiting to write.
The maximum wait times are:

DISPLAY MODE	MAXIMUM WAITING TIME
H32cell	APPROXIMATE 5.96μsec
H40cell	APPROXIMATE 4.77μsec

As the CPU has unlimited access to the RAMs during vertical blanking, the wait case never arises.

◇ HV COUNTER ◇

The HV counter's function is to give the horizontal and vertical location of the television beam. If the "M3" bit of REGISTER #0 is set, the HV counter will then freeze when trigger signal HL goes high, as well as triggering a level 2 interrupt.

M3	COUNTER LATCH MODE
0	COUNTER IS NOT LATCHED BY TRIGGER SIGNAL
1	COUNTER IS LATCHED BY TRIGGER SIGNAL

M3: REGISTER#0

NON INTERLACE MODE

$C00008									
	VC7	VC6	VC5	VC4	VC3	VC2	VC1	VC0	(D15~ D8)
	HC8	HC7	HC6	HC5	HC4	HC3	HC2	HC1	(D7 ~ D0)

INTERLACE MODE

$C00008									
	VC7	VC6	VC5	VC4	VC3	VC2	VC1	VC8	(D15~ D8)
	HC8	HC7	HC6	HC5	HC4	HC3	HC2	HC1	(D7 ~ D0)

V-COUNTER : VC7~VC0

DISPLAY MODE	COUNTER DATA
V 28 CELL	0~ DFH
V 30 CELL	0~ EFH

H-COUNTER : HC8~HC1

DISPLAY MODE	COUNTER DATA
H 32 CELL	0~ 7FH
H 40 CELL	0~ 9FH

The counter only has eight bits each for H and V, so interlace mode and 40 cell(320 dot) modes present some problems. DUring interlace mode, the LSB of the vertical position is replaced by the new MSB. And the horizontal resolution problem is solved by ALWAYS dropping the LSB.

CAUTION:
As the HV counter's value is not valid during vertical blanking, check to be sure that it is active scan before using the value.

§7 DMA TRANSFER

DMA (Direct Memory Access) is a high speed technique for memory accesses to the VRAM, CRAM and VSRAM. During DMA VRAM, CRAM and VSRAM occur at the fastest possible rate (please see the section on Access Timing). There are three modes of DMA access, as can be seen below, all of which may be done to VRAM or CRAM or VSRAM. The 68K is stopped during memory to VRAM/CRAM/VSRAM DMA, but the Z80 countinues to run as long as it does not attempt access to the 68K memory space.
The DMA is quite fast during VBLANK, about double the tightest possible 68K loop's speed, but during active scan the speed is the same as a 68K loop.
Please note that after this point, VRAM is used as a generic term for VRAM/CRAM/VSRAM.

DMD1	DMD0	DMA MODE	SIZE
0	SA23	A. MEMORY TO V-RAM	WORD to BYTE(H)&(L)
1	0	B. VRAM FILL	BYTE to BYTE
1	1	C. VRAM COPY	BYTE to BYTE

DMD1, DMD0: REG#23 ※ DMD0=SA23

Source address are \$000000-\$3FFFFF (ROM) and \$FF0000~\$FFFFFF (RAM) for memory to VRAM transfers. In the case of ROM to VRAM transfers, a hardware feature causes occasional failure of DMA unless the following two conditions are observed:

- --The destination address write (to address \$C00004) must be a word write
- --The final write must use the work RAM.

There are two ways to accomplish this, by copying the DMA program into RAM or by doing a final "move.w ram address, \$C00004".

◇ MEMORY TO VRAM ◇

The function transfers data from 68K memory to VRAM, CRAM or VSRAM. During this DMA all 68K processing stops. The source address is \$000000-\$3FFFFF for ROM or \$FF0000-\$FFFFFF for RAM. The DMA reads are word wide, writes are byte wide for VRAM and word wide for CRAM and VSRAM. The destination is specifed by:

CD2	CD1	CD0	MEMORY TYPE
0	0	1	VRAM
0	1	1	CRAM
1	0	1	VSRAM

Setting of DMA

(A) M1 (REG. #1) = 1 : DMA ENABLE
(B) Increment No. set to #15 (normally 2)
(C) Transfer word No. set into #19, #20.
(D) Source address and DMA mode set into #21, #22, #23.
(E) Set the destination address.
(F) ☆VDP gets the CPU bus.
(G) ☆DMA start.
(H) ☆VDP releases the CPU bus.
(I) M1 have to be 0 after confirmation of DMA finish : DMA DISENABLE

DMA starts after (E).
You must set M1=1 only durilng DMA otherwise we cannot quarantee the operation. Souarce address were increased with +2 and destination address increased with content of register #15.

Content ; of register. Register #1 has another bits.

REG. #15	INC7	INC6	INC5	INC4	INC3	INC2	INC1	INC0

INC7～INC0 : No. of increment

REG. #1	0	DISP	IE0	M1	M2	1	0	0	
#19	LG7	LG6	LG5	LG4	LG3	LG2	LG1	LG0	
#20	LG15	LG14	LG13	LG12	LG11	LG10	LG9	LG8	
#21	SA8	SA7	SA6	SA5	SA4	SA3	SA2	SA1	
#22	SA16	SA15	SA14	SA13	SA12	SA11	SA10	SA9	
#23	0	SA23	SA22	SA21	SA20	SA19	SA18	SA17	
1st $C00004	CD1	CD0	DA13	DA12	DA11	DA10	DA9	DA8	(D15～ D8)
	DA7	DA6	DA5	DA4	DA3	DA2	DA1	DA0	(D7 ～ D0)
2nd $C00004	0	0	0	0	0	0	0	0	(D15～ D8)
	1	0	0	CD2	0	0	DA15	DA14	(D7 ～ D0)

LG15～LG0 : No. of move word
SA23～SA1 : Source address (in 68000)
DA15～DA0 : Destination address (in VDP)
CD2～CD0 : RAM selection

◇ VRAM FILL ◇

FILL mode filles with same data from free even VRAM address. FILL for only VRAM.

How to set FILL(DMA).

(A) M1(REG.#1)=1 : DMA ENABLE
(B) Increment No. set to #15(normally 1).
(C) Fill size set to #19, #20.
(D) DMA mode set to #23.
(E) Distination address and FILL data set.
(F) ☆DMA start
(G) M1=0 after confirmation of finishing
: DMA DISENABLE

DMA starts at after (E).
M1 should be 1 in the DMA transfer otherwith we cannot quarantee the operation.
Distination address is incremented with register #15. VDP dose not asks bus open for CPU but CPU cannot access VDP without PSG, HV counter and status. You can realize enf of DMA by DMA bit in status register.

Register setting. Register#1 has another bits.

REG. #15

INC7	INC6	INC5	INC4	INC3	INC2	INC1	INC0

INC7~INC0 : Increment No.

STATUS

*	*	*	*	*	*	EMPT	FULL
F	SOVR	C	ODD	VB	HB	DMA	PAL

DMA : 1: DMA BUSY
* : Not care

REG. #1	0	DISP	IE0	M1	M2	1	0	0	
#19	LG7	LG6	LG5	LG4	LG3	LG2	LG1	LG0	
#20	LG15	LG14	LG13	LG12	LG11	LG10	LG9	LG8	
#23	1	0	0	0	0	0	0	0	
1st $C00004	0	1	DA13	DA12	DA11	DA10	DA9	DA8	(D15~ D8)
	DA7	DA6	DA5	DA4	DA3	DA2	DA1	DA0	(D7 ~ D0)
2nd $C00004	0	0	0	0	0	0	0	0	(D15~ D8)
	1	0	0	0	0	0	DA15	DA14	(D7 ~ D0)
$C00000	FD15	FD14	FD13	FD12	FD11	FD10	FD9	FD8	(D15~ D8)
	FD7	FD6	FD5	FD4	FD3	FD2	FD1	FD0	(D7 ~ D0)

LG15~LG0 : FILL byte No.
DA15~DA0 : Destination address
FD15~FD0 : FILL data

When setting 1st and 2nd by long word, 1st will be D31-D16 and 2nd, D15-D0.

EXAMPLE

1 TERM: FILL data are word; register #15=1

a. V-RAM address is even.

(A) First, low side of FILL data are written in V-RAM address.
(B) Second, upper side of FILL data are written in V-RAM+1.
(C) And, V-RAM address is added register #15, written upper side FILL data in V-RAM at next each step.

b. V-RAM address is odd.

(D) First, upper side of FILL data are written in V-RAM addresss-1.
(E) Seconf, low side of FIll data are written in V-RAM.
(F) Same as (C).

·VRAM address is even;

ADD	(A)	Even
ADD+1	(B) (C)	Odd
ADD+2	(C)	
ADD+3	(C)	
ADD+4	(C)	
ADD+5	(C)	
ADD+6	(C)	
ADD+7		

·VRAM address is odd;

ADD-1	(D)	Even
ADD	(E)	Odd
ADD+1	(F)	
ADD+2	(F)	
ADD+3	(F)	
ADD+4	(F)	
ADD+5	(F)	
ADD+6	(F)	
ADD+7		

※ You must rewrite data (C) into ADD+1 after write data (B).

2 TERM: FILL data are word; register #15=2

· VRAM address=even

ADD	(A)lower	Even
ADD+1	(B)upper	Odd
ADD+2	(C)lower	
ADD+3	upper	
ADD+4	(C)lower	
ADD+5	upper	
ADD+6	(C)lower	
ADD+7	upper	

· VRAM address=odd

ADD-1	(D)upper	Even
ADD	(E)lower	Odd
ADD+1		
ADD+2	(F)upper	
ADD+3	lower	
ADD+4	(F)upper	
ADD+5	lower	
ADD+6	(F)upper	
ADD+7	lower	

3 TERM: FILL data are byte.

a. V-RAM address is even.

(A) = (B) = (C) =BYTE·DATA

b. V-RAM address is odd.

(D) = (E) = (F) =BYTE·DATA

◇ VRAM COPY ◇

This function dose copy from source address to destination address by number of COPY byte.

DMA setting

(A) M1 (REG. #1) = 1 : DMA ENABLE
(B) Number of copy bytes in #19, #20
(C) Source address and DMA mode in #23.
(D) Destination address set.
(E) ☆DAM transfer
(F) After confirming DMA finish: M1=0
 : DMA DISENABLE

DMA starts when (D) above is finished. Apply M1=1 only during DMA transfer. In other cases, if M1=1 is set, there is no quaranty that it will function correctly. At the time of DMA transfer, the destination address is incremented by the set value of REG. #15. During DMA transfer, although the VDP does not require CPU to maka a bus available, no access is possible from CPU to VDP except for PSG, HV counter, STATUS READ. DMA transfer finish can be recognized by referring to the STATUS REGISTER's DMA bit.

Example: With TRANSFER BYTE=3 at the time of VRAM COPY

SOURCE ADDRESS	REG#15=1 DESTINATION ADRESS	REG#15=2 DESATINATION ADDRESS
DATA1	DATA1	DATA1
DATA2	DATA2	
DATA3	DATA3	DATA2
DATA4		
DATA5		DATA3
DATA6		
DATA7		

※ CAUTION

In the case of VRAM COPY, "read form VRAM" and "write ato VRAM" are repeated perbyte. Therefore, when the SOURCE AREA and TRANSFER AREA are over lapped, the transfer may not be performed correctly.

REGISTER are as follows. REGISTER #1 includes bits set for purposes other than DMA. Therefore, pay careful attention in this regard.

REG. #15	INC7	INC6	INC5	INC4	INC3	INC2	INC1	INC0

INC7~INC0 : Increment No.

STATUS	*	*	*	*	*	*	EMPT	FULL
	F	SOVR	C	ODD	VB	HB	DMA	PAL

DMA : 1: DMA BUSY

REG. #1	0	DISP	IE0	M1	M2	1	0	0	
#19	LG7	LG6	LG5	LG4	LG3	LG2	LG1	LG0	
#20	LG15	LG14	LG13	LG12	LG11	LG10	LG9	LG8	
#21	SA7	SA6	SA5	SA4	SA3	SA2	SA1	SA0	
#22	SA15	SA14	SA13	SA12	SA11	SA10	SA9	SA8	
#23	1	1	0	0	0	0	0	0	
1st $C00004	0	0	DA13	DA12	DA11	DA10	DA9	DA8	(D15~ D8)
	DA7	DA6	DA5	DA4	DA3	DA2	DA1	DA0	(D7 ~ D0)
2nd $C00004	0	0	0	0	0	0	0	0	(D15~ D8)
	1	1	0	0	0	0	DA15	DA14	(D7 ~ D0)

LG15~LG0 : Number of copy byte
SA15~SA0 : Source address
DA15~DA0 : Destination address

When setting 1st and 2nd by long word, 1st will be D31~D16 and 2nd, D15~D0.

◇ DMA TRANSFER CAPACITY ◇

Transfer quantity varies depending on the DISPLAY MODE as follows:

DMA MODE	DISPLAY MODE	SCREEN SCANNING	TRANSFER BYTES PER LINE
MEMORY TO VRAM	H32CELL	DURING EFFECTIVE SCREEN DURING V BLANK	16Bytes 167Bytes
	H40CELL	DURING EFFECTIVE SCREEN DURING V BLANK	18Bytes 205Bytes
VRAM FILL	H32CELL	DURING EFFECTIVE SCREEN DURING V BLANK	15Bytes 166Bytes
	H40CELL	DURING EFFECTIVE SCREEN DURING V BLANK	17Bytes 204Bytes
VRAM COPY	H32CELL	DURING EFFECTIVE SCREEN DURING V BLANK	8Bytes 83Bytes
	H40CELL	DURING EFFECTIVE SCREEN DURING V BLANK	9Bytes 102Bytes

In the MEMORY TO VRAM, in the case where CRAM and VSRAM are the desstinations, number of words (not byte) should apply. One line during V BLANK allows for data transfer to all the address of CRAM and VSRAM.

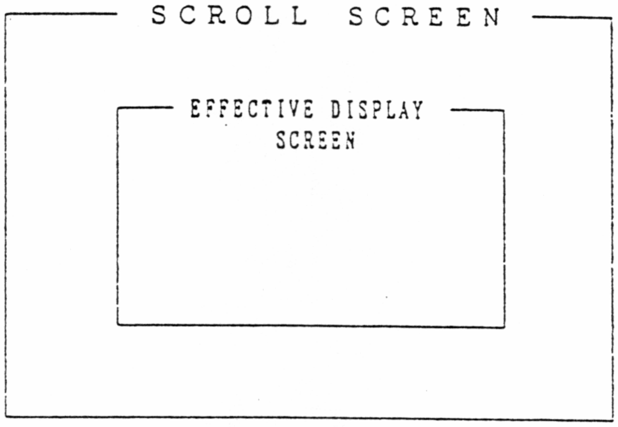
Note that when calculating, hte transfer quantity in one screen (1/60 sec) varies depending on the number of LINES during V BLANK (refer to DISPLAY MODE) in the case of NTSC (video signal) and PAL systems.

DISPLAY MODE	NO.OF HORIZONTAL LINE
V28CELL (NTSC)	36
V28CELL (PAL)	87
V30CELL (PAL)	71

Where REGISTER #1 DISP=0, i. e., when on-screen display is not made, the TRANSFER quantity is the same aas TRANSFER BYTES PER LINE during BLANKING.

§8 SCROLLING SCREEN

There are two different scroll screens, i. e., A and B which separately can scroll vertically and horizontally on a basis of a one a one dot unit. In the horizontal direction, scrolling overall or based on a one cell unit or one line unit can be selected, and in the vertical direction , scrolling overall or in a two cell unit can be selected. Also, the scroll screen size can be changed on a basis of a 32 cell unit.



For the scrolling screen display, the following REGISTER setting and VRAM ara are required.

O SCROLL "A" PATTERN NAME TABLE BASE ADDRESS

REG. #2	0	0	SA15	SA14	SA13	0	0	0

O SCROLL "B" PATTERN NAME TABLE BASE ADDRESS

REG. #4	0	0	0	0	0	SB15	SB14	SB13

O MODE SET REGISTER No. 3

REG. #11	0	0	0	0	IE2	VSCR	HSCR	LSCR

O MODE SET REGISTER No. 4

REG. #12	RS0	0	0	0	S/TE	LSM1	LSM0	RS1

O H SCROLL DATA TABLE BASE ADDRESS

REG. #13	0	0	HS15	HS14	HS13	HS12	HS11	HS10

O SCROLL SIZE

REG. #16	0	0	VSZ1	VSZ0	0	0	HSZ1	HSZ0

O VRAM	:	SCROLL"A" PATTERN NAME TABLE	max8KByte
		SCROLL"B" PATTERN NAME TABLE	max8KByte
		H SCROLL DATA TABLE	max960Byte
O VSRAM	:	V SCROLL DATA TABLE	max80Byte

◇ SCROLLING SCREEN SIZE ◇

The screen size can be set by VSZ1, VSZ0, HSZ1, and HSZ0 (REG. #16). The following 6 kinds can be set both for SCROLL SCREEN A and B.

32*32/32*64/32*128
64*32/64*64
128*32

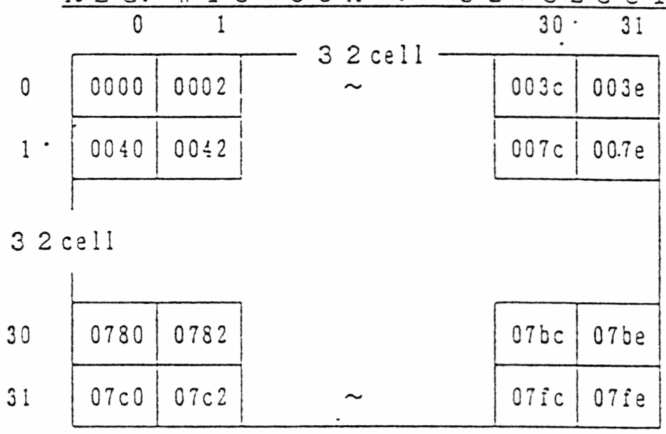
VSZ1	VSZ0	FUNCTION
0	0	V 32cell
0	1	V 64cell
1	0	PROHIBITED
1	1	V128cell

HSZ1	HSZ0	FUNCTION
0	0	H 32cell
0	1	H 64cell
1	0	PROHIBITED
1	1	H128cell

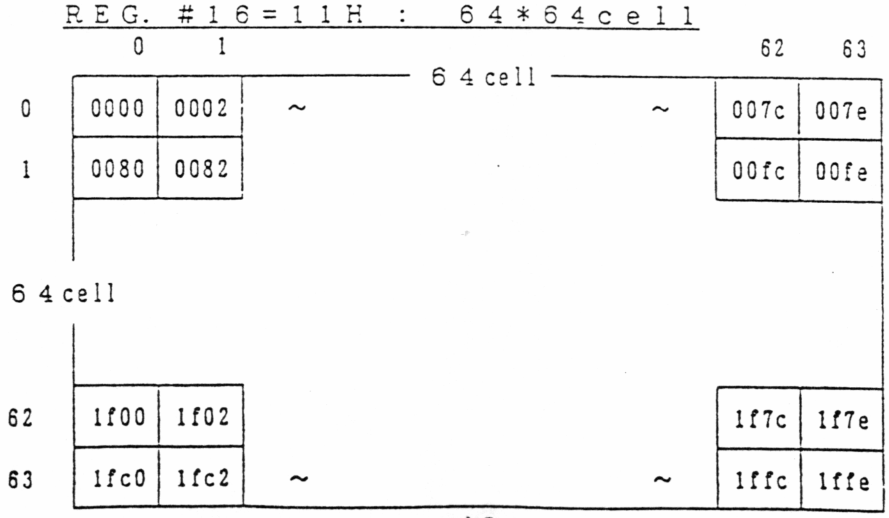
SCROLL SCREEN's PATTERN NAME TABLE ADDRESS exits in the VRAM and is designated by REGSTER #2 and #4. Depending VRAM and SCROLL screen correspond to each other differently.

EXAMPLE

REG. #16=00H : 32*32cell



REG. #16=11H : 64*64cell



REG. #16=03H : 32*128cell

128cell (columns 0–127), 32cell (rows 0–31)

	0	1	~	126	127
0	0000	0002	~	00fc	00fe
1	0100	0102		01fc	01fe
30	1e00	1e02		1efc	1efe
31	1f00	1f02	~	1ffc	1ffe

A Value shown in a frame indicates an offset from the PATTERN NAME TABLE BASE ADDRESS.

◇HORIZONTAL SCROLLING◇

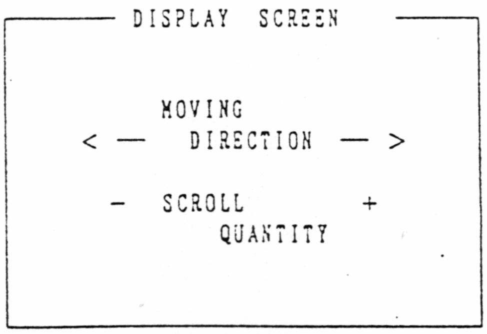
The DISPLAY SCREEN allows for scrolling overall, or based on one cell unit, or on an dot by dot basis in one line unit. Either one of the above scrolling can be selected by HSCR and LSCR(REGSTER#11). A setting appleis to both SCROLL screen A and B.

HSCR	LSCR	FUNCTION
0	0	OVERALL SCROLLING
0	1	PROHIBITED
1	0	SCROLL IN ONE CELL UNIT
1	1	SCROLL IN ONE LINE UNIT

HSCR, LSCR: REG. #11

The effective scroll quantity is equivalent to 10 bits(000H~3FFH).

Taking the DISPLAY SCREEN as standard, the SCROLL direction will be as follows:



Horizontally scrolling quantity setting area: H Scroll DATA TABLE is in VRAM. From the base address which was set by REG. 13, set the scrolling quantity of SCREEN A and B alternately. Also, the scrolling quantity data setting position waries dependning on the following mode (OVERALL, 1 cell, or 1 line).

MODE	SETTING POSITION
OVERALL	LINE 0
1 CELL	EVERY 8TH LINE STARTING FROM LINE 0
1 LINE	ALL LINES

	15–10	9–0	
00		A·SCROLLING QUANTITY OF SCREEN A	OVERALL, CELL, LINE
02		B·SCROLLING QUANTITY OF SCREEN B	OVERALL, CELL, LINE
04		A·SCROLLING QUANTITY OF SCREEN A	LINE
06		B·SCROLLING QUANTITY OF SCREEN B	LINE
08		A·SCROLLING QUANTITY OF SCREEN A	LINE
0A		B·SCROLLING QUANTITY OF SCREEN B	LINE
1C		A·SCROLLING QUANTITY OF SCREEN A	LINE
1E		B·SCROLLING QUANTITY OF SCREEN B	LINE
20		A·SCROLLING QUANTITY OF SCREEN A	CELL, LINE
22		B·SCROLLING QUANTITY OF SCREEN B	CELL, LINE
3FC		A·SCROLLING QUANTITY OF SCREEN A	LINE
3FE		B·SCROLLING QUANTITY OF SCREEN B	LINE

D15~D10 can be greely utilized for program software.

◇ V SCROLL ◇

The DISPLAY SCREEN allows for scrolling overall or every 2 CELLs in a dot unit. The setting can be done by VSCR (REG. #11). A setting applies to both SCROLL SCREEN A and B.

VSCR	FUNCTION
0	OVERALL SCROLL
1	2-CELL UNIT SCROLL

VSCR: REG. #11

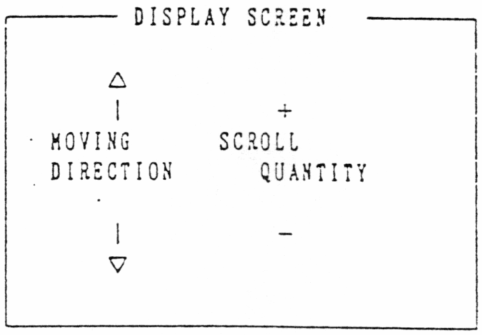
The scrolling quantity is equivalent to 11bits (000H~7FFH). However, it will be as shown below in the INTERLACE MODE.

NONINTERLACE: The effective scrolling quantity is equivalent to 10 bits.

INTERLACE 1: -ditto-

INTERLACE 2: The effective scrolling quantity is equivalent to 11 bits.

Taking the DISPLAY SCREEN as standard, the scrolling direction will be as follows:



Set the V SCROLL quantity by VSRAM.
Alternately set the SCroll quantity of SCREEN A and B.
Depending on the SCROLL MODE, the DATA setting positions differ.

MODE	SETTING POSITION
OVERALL	ONLY AT THE BEGINNING
2-CELL	SET TO ALL

	15 14 13 12 11	10 9 8 7 6 5 4 3 2 1 0	
00		A:SCROLL QUANTITY OF SCREEN A	0,1 CELL,OVERALL
02		B:SCROLL QUANTITY OF SCREEN B	0,1 CELL,OVERALL
04		A:SCROLL QUANTITY OF SCREEN A	2,3 CELL
06		B:SCROLL QUANTITY OF SCREEN B	2,3 CELL
08		A:SCROLL QUANTITY OF SCREEN A	4,5 CELL
0A		B:SCROLL QUANTITY OF SCREEN B	4,5 CELL
0C		A:SCROLL QUANTITY OF SCREEN A	6,7 CELL
0E		B:SCROLL QUANTITY OF SCREEN B	6,7 CELL
4C		A:SCROLL QUANTITY OF SCREEN A	38,39 CELL
4E		B:SCROLL QUANTITY OF SCREEN B	38,39 CELL

D15~D11 is indefinite.

◇ SCROLL PATTERN NAME ◇

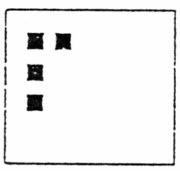
The SCROLL SCREEN's name table is in VRAM and set by REG. #2 and #4. The PATTERN NAME requires 2 bytes(1word) per CELL the SCROLL screen. Depending on the SCROLL screen's size, VRAM and SCROLL SCREEN correspond with each other differntly. Refer to SCROLL SCREEN SIZE.

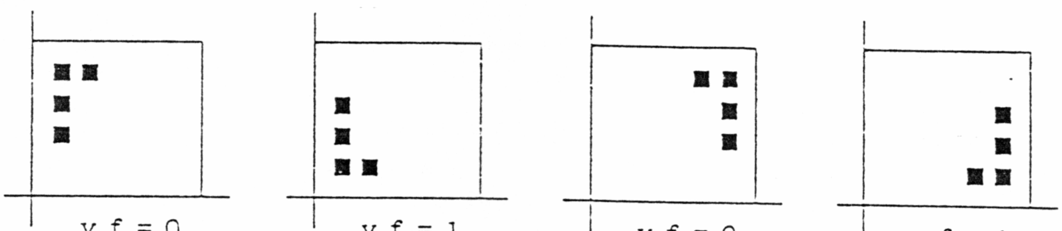
PATTERN NAME									
	pri	cp1	cp0	vf	hf	pt10	pt9	pt8	(d15~ d8)
	pt7	pt6	pt5	pt4	pt3	pt2	pt1	pt0	(d7 ~ d0)

pri : Refer to PRIORITY
cp1 : Color palette selection bit
(See COLOR PALETTE)
cp0 : -ditto-
vf : V REVERSE bit 1: REVERSE
hf : H REVERSE bit 1: REVERSE

pt10~pt0 : PATTERN GENERATOR NUMBER

REVERSE BIT vf and hf : Allows for H and V reverse on CELL unit basis.





vf = 0 hf = 0	vf = 1 hf = 0	vf = 0 hf = 1	vf = 1 hf = 1

◇PATTERN GENERATOR◇

PATTERN GENERATOR has VRAM 0000H as base address, and a pattern is expressed on a 8×8 dot bais. To define a pattern, 32 bytes are required. Starting from 0000H, it proceeds in the sequence of PATTERN GENERATOR 0, 1, 2, …. The relationship between the display pattern and memory is as follows:

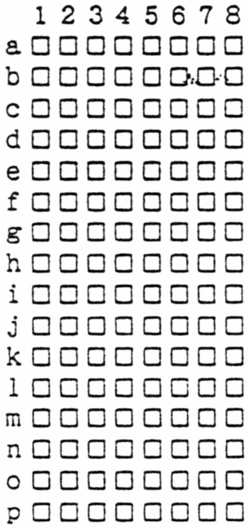
```
  1 2 3 4 5 6 7 8
a □ □ □ □ □ □ □ □
b □ □ □ □ □ □ □ □
c □ □ □ □ □ □ □ □
d □ □ □ □ □ □ □ □
e □ □ □ □ □ □ □ □
f □ □ □ □ □ □ □ □
g □ □ □ □ □ □ □ □
h □ □ □ □ □ □ □ □
```

	0		1		2		3	
	D7	D0	D7	D0	D7	D0	D7	D0
00	a 1	a 2	a 3	a 4	a 5	a 6	a 7	a 8
04	b 1	b 2	b 3	b 4	b 5	b 6	b 7	b 8
08	c 1	c 2	c 3	c 4	c 5	c 6	c 7	c 8
0C	d 1	d 2	d 3	d 4	d 5	d 6	d 7	d 8
10	e 1	e 2	e 3	e 4	e 5	e 6	e 7	e 8
14	f 1	f 2	f 3	f 4	f 5	f 6	f 7	f 8
18	g 1	g 2	g 3	g 4	g 5	g 6	g 7	g 8
1C	h 1	h 2	h 3	h 4	h 5	h 6	h 7	h 8

The display colors and memory relationship is as follows:

D7	D6	D5	D4	D3	D2	D1	D0
COL3	COL2	COL1	COL0	COL3	COL2	COL1	COL0

In INTERLACE MODE 2, one cell consists of 8×16 dots and therefore, 64 Bytes (16 long words) are required.



	0		1		2		3	
	D7	D0	D7	D0	D7	D0	D7	D0
00	a1	a2	a3	a4	a5	a6	a7	a8
04	b1	b2	b3	b4	b5	b6	b7	b8
08	c1	c2	c3	c4	c5	c6	c7	c8
0C	d1	d2	d3	d4	d5	d6	d7	d8
10	e1	e2	e3	e4	e5	e6	e7	e8
14	f1	f2	f3	f4	f5	f6	f7	f8
18	g1	g2	g3	g4	g5	g6	g7	g8
1C	h1	h2	h3	h4	h5	h6	h7	h8
20	i1	i2	i3	i4	i5	i6	i7	i8
24	j1	j2	j3	j4	j5	j6	j7	j8
28	k1	k2	k3	k4	k5	k6	k7	k8
2C	l1	l2	l3	l4	l5	l6	l7	l8
30	m1	m2	m3	m4	m5	m6	m7	m8
34	n1	n2	n3	n4	n5	n6	n7	n8
38	o1	o2	o3	o4	o5	o6	o7	o8
3C	p1	p2	p3	p4	p5	p6	p7	p8

§9 WINDOW

For WINDOW display, the following register setting and VRAM areas are required.

○ WINDOW PATTERN NAME TABLE AND BASE ADDRESS

REG. #3	0	0	WD15	WD14	WD13	WD12	WD11	0

○ MODE SET REGISTER No. 4

REG. #12	RS0	0	0	0	S/TE	LSM1	LSM0	RS1

○ WINDOW H POSITION

REG. #17	RIGT	0	0	WHP5	WHP4	WHP3	WHP2	WHP1

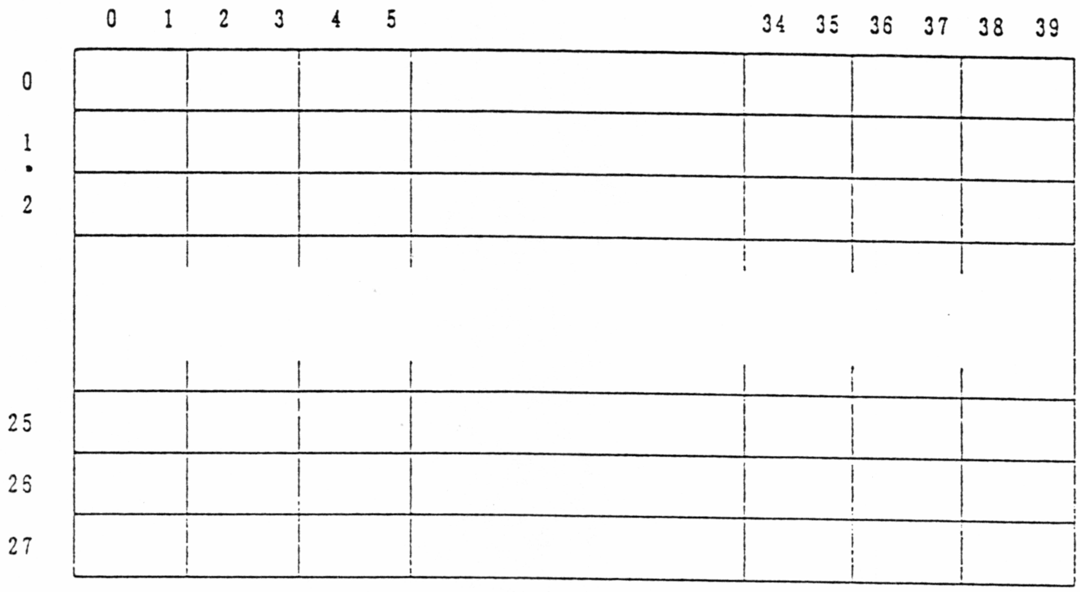
○ WINDOW V POSITION

REG. #18	DOWN	0	0	WVP4	WVP3	WVP2	WVP1	WVP0

○ VRAM: WINDOW PATTERN NAME TABLE MAX 4K BYTES

◇ DISPLAY POSITION ◇

The WINDOW DISPLAY POSITION is designated by REG. #17 and #18.
Screen display can be divided on a unit basis of H 2 cells and V 1 cell. The dividing position varies depending on resolution.



H 40 CELLS/V 28 CELLS MODE

REG. #17

RIGT	0	0	WHP5	WHP4	WHP3	WHP2	WHP1

REG. #18

DOWN	0	0	WVP4	WVP3	WVP2	WVP1	WVP0

RIGT : 0 Displays WINDOW from the lefted to H dividing position.
1 Displays WINDOW from the H dividing position to the right end.

DOWN : 0 Displays WINDOW from the top end to the V dividing position.
1 Displays WINDOW from the V dividing position to the bottom end.

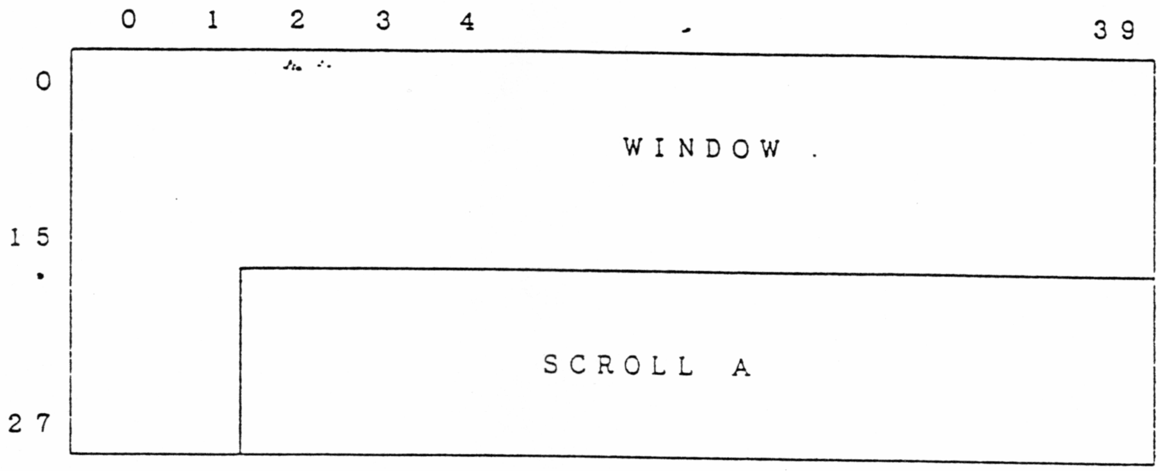
WHP5～WHP1 : H dividing position
WVP4～WVP0 : V dividing position

H RESOLUTION	DIVIDING POSITION(WHP)
32 CELL	0～16 (0～32CELL)
40 CELL	0～20 (0～40CELL)

V RESOLUTION	DIVIDING POSITION(WVP)
28 CELL	0～28
30 CELL	0～30

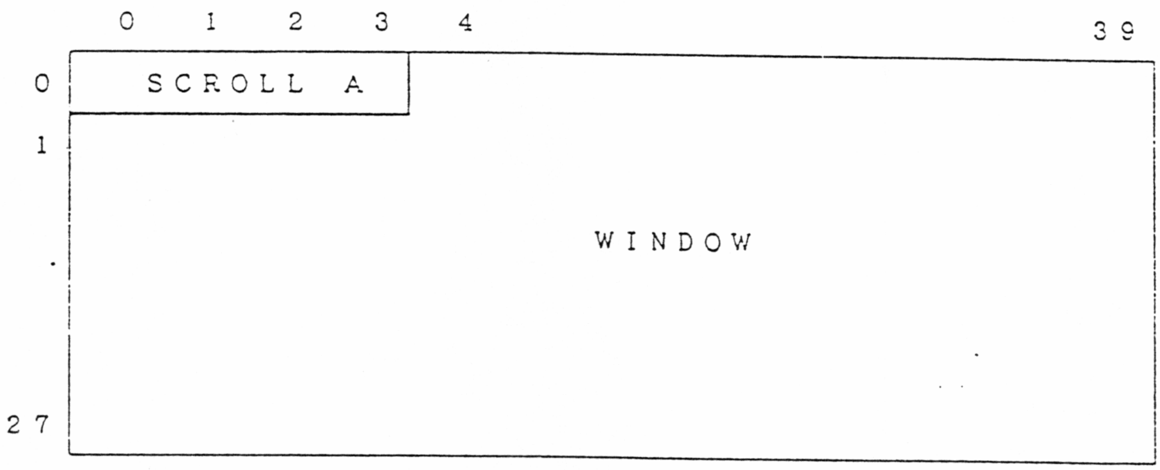
SETTING EXAMPLE

REG. #17 : 00H+01H WINDOW from the left end to the second cell
REG. #18 : 00H+10H WINDOW from the top end to the 16th cell



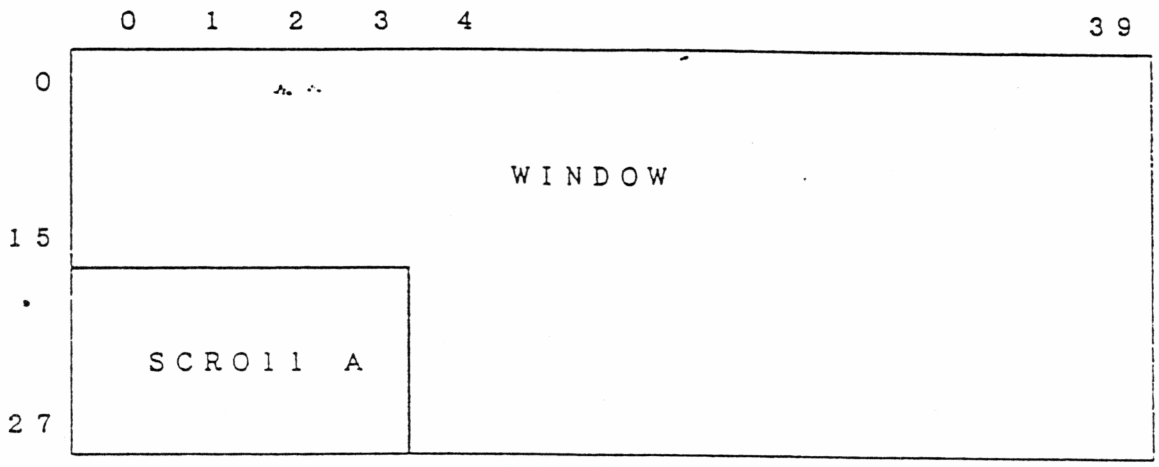
DISPALY SCREEN:40×28 CELL MODE

REG. #17 : 80H+02H WINDOW from the left end 4th cell to the right end
REG. #18 : 80H+01H WINDOW from the 2nd cell to the bottom end



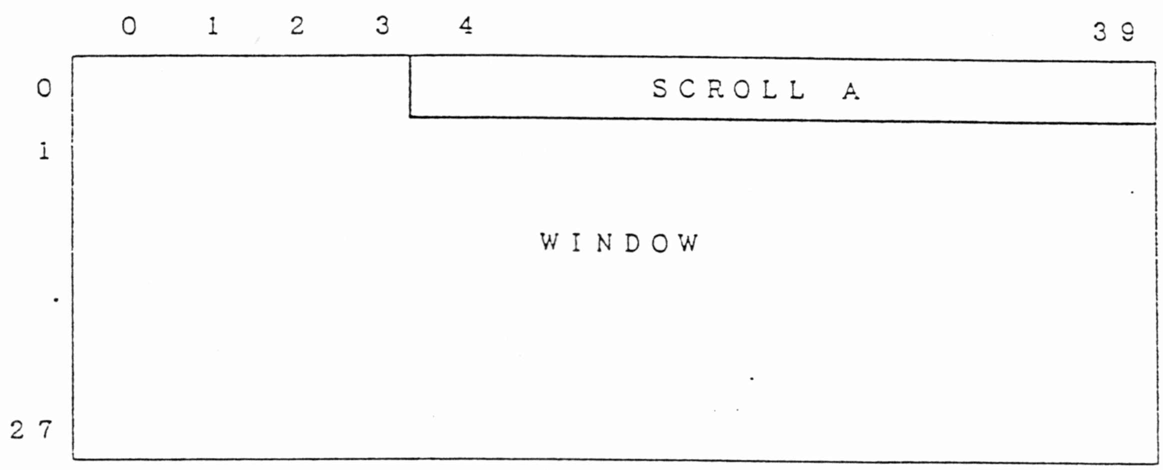
DISPALY SCREEN:40×28 CELL MODE

REG. #17 : 80H+01H WINDOW from the 4th cell to the right end
REG. #18 : 00H+10H WINDOW from the top end to the 16th cell



DISPALY SCREEN:40×28 CELL MODE

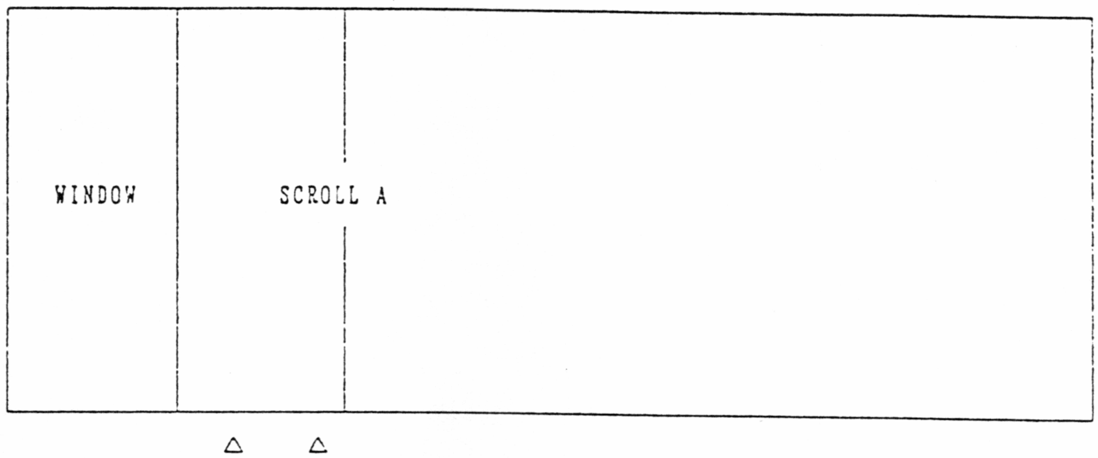
REG. #17 : 00H+02H WINDOW to the 4th cell from the left
REG. #18 : 80H+01H WINDOW from the 2nd cell to the bottom end



DISPALY SCREEN:40×28 CELL MODE

◇ WINDOW PRIORITY ◇

WINDOW PRIORITY is handled in the same way as in SCROLL A. SCROLL A is not displayed in the area where WINDOW is displayed. Also, only when WINDOW is set to the left and SCROLL A is moved in H direction, the character corresponding to 2 cells on the right side of the boundary between WINDOW and SCROLL A will be disfigured.
There will be no malfunvtioning when WINDOW is set to the left side and SCROLL A is moved only in V direction, and also when WINDOW is set to the right side.



Display of this portion will be disfigured,therefore,mask SCROLL A by using high priiority.

◇ WINDOW PATTERN NAME ◇

WINDOW PATTERN NAME TABLE is on VRAM, and the BASE ADDRESS is designated by REG. #13.
The PATTERN NAME, the same as in SCROLL SCREEN, requires 2 bytes(1word) per cell.

PATTERN NAME									
	pri	cp1	cp0	vf	hf	pt10	pt9	pt8	(d15～ d8)
	pt7	pt6	pt5	pt4	pt3	pt2	pt1	pt0	(d7 ～ d0)

pri : Refer to PRIORITY
cp1 : Color Palette Selection bit
cp0 : -ditto-
vf : V REVERSE BIT 1: REVERSE
hf : H REVERSE BIT 1: REVERSE

pt10～pt0 : PATTERN GENERATOR NO.

PATTERN NAME and VRAM relation varies depending on H 32 cell/40 cell mode. Pay careful attention to this point.

H 32 CELL MODE

(32 cell across; 32 cell down)

	0	1		30	31
0	0000	0002	~	003c	003e
1	0040	0042		007c	007e
⋮					
30	0780	0782		07bc	07be
31	07c0	07c2	~	07fc	07fe

H 40 CELL MODE

40～63 are not displayed

(32 cell down)

	0	1		39▽	40		62	63
0	0000	0002	~	004e	0050	~	007c	007e
1	0080	0082		00dc	00e0		00fc	00fe
⋮								
30	0f00	0f02		0f4e	0f50		0f7c	0f7e
31	0fc0	0fc2	~	0fde	0fe0	~	0ffc	0ffe

Values shown are offset from the BASE ADDRESS

In the H 40cell mode, there exists the area for H 64cells. However, there will be no display from the 41st cell in the H direction.

Also, in the V 28cell mode, there will be no display from V 29th cell; and in the 30th cell mode, there will be no display from 31st cell.

§10 SPRITE

For sptite display, the following REGISTER setting and VRAM area are required.

○ SPRITE ATTRIBUTE TABLE AND BASE ADDRESS

REG. #5							
0	AT15	AT14	AT13	AT12	AT11	AT10	AT9

○ MODE SETTING REGISTER No. 4

REG. #12							
RS0	0	0	0	S/TE	LSM1	LSM0	RS1

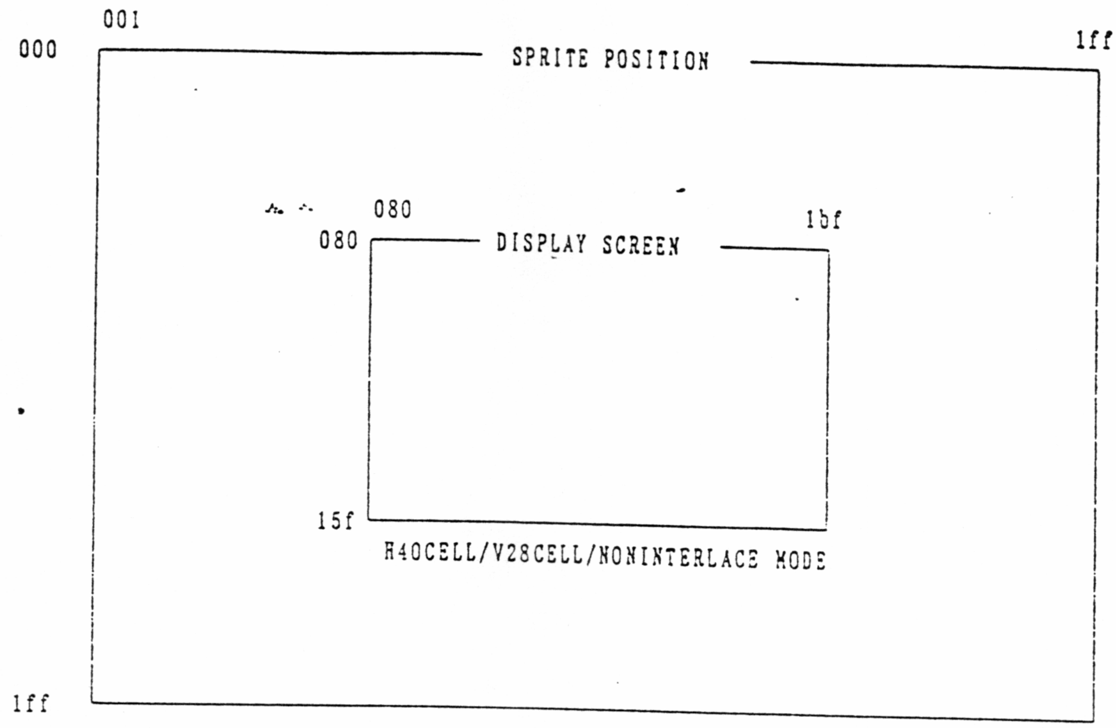
○ VRAM: SPRITE ATTRIBUTE TABLE MAX640BYTES

◇ DISPLAY POSITION ◇

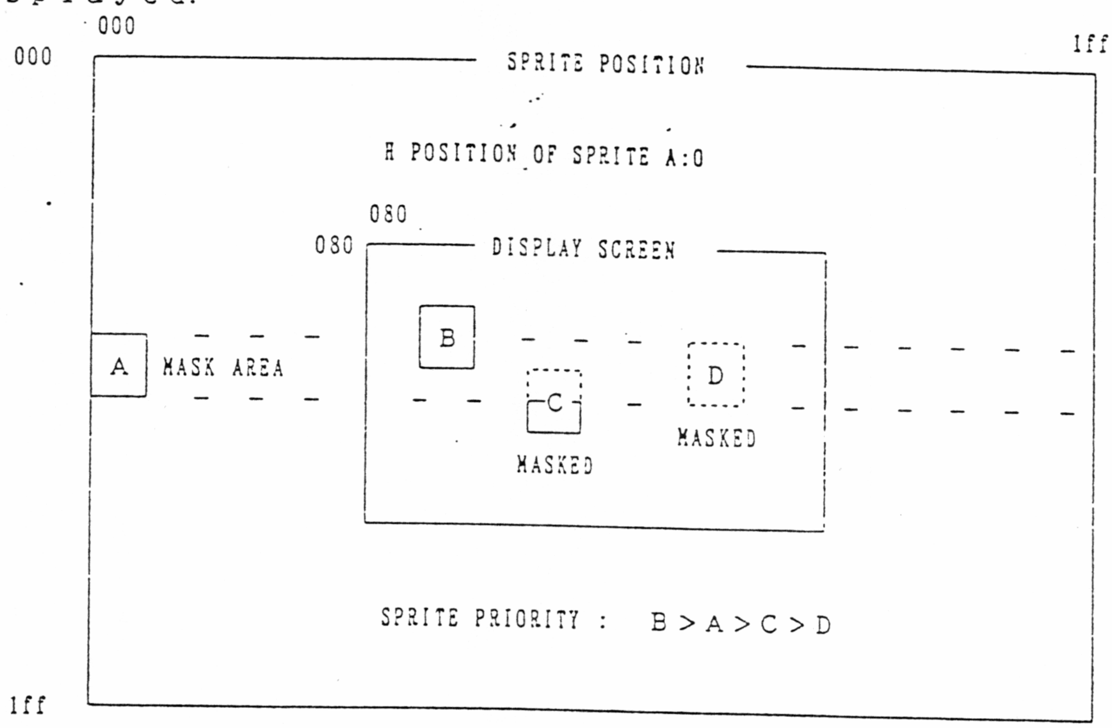
SPRITE POSITION and DISPLAY SCREEN are as follows: When sprite H position is 0, this is in a special mode, therefore, pay careful attention to this point.

RESOLUTION	H POSITION	DISPLAY AREA
H 32CELL	001~1FFH	080~17FH
H 40CELL		080~1BFH

RESOLUTION		V POSITION	DISPLAY AREA
V 28CELL	NONINTERLASCE	000~1FFH	080~15FH
	INTERLACE 1		
	INTERLACE 2	000~3FFH	100~2BFH
V 30CELL	NONINTERLACE	000~1FFH	080~16FH
	INTERLACE 1		
	INTERLACE 2	000~3FFH	100~2DFH



When 0 is set to the SPrite H POSITION, a low priority sprite on the same line will not be displayed.



◇SPRITE ATTRIBUTE◇

SPRITE ATTRIBUTE TABLE is on VRAM and the BASE ADDRESS is designated by REG. #5. ATTRIBUTE requires 8bytes(4words) per SPRITE, and indicates DISPLAY POSITION, PRIORITY, SPRITE GENERATOR Number and ATTRIBUTE.

Starting from the beginning of the ATTRIBUTE TABLE, numbers are given in the sequence of SPRITE 0, SPRITE 1, SPRITE 2, SPRITE 3, …. Priority between SPRITEs is not determined by the sequence of sprite number, but by each SPRITE's LINK DATA, and thus becomes programmable.

1st
VPOSITION

						vp9	vp8	(d15~ d8)
vp7	vp6	vp5	vp4	vp3	vp2	vp1	vp0	(d7 ~ d0)

2nd
SPRITE SIZE
LINK DATA

				hs1	hs0	vs1	vs0	(d15~ d8)
	ld6	ld5	ld4	ld3	ld2	ld1	ld0	(d7 ~ d0)

3rd
PRIORITY, PALETTE, REVERSE, PATTERN

pri	cp1	cp0	vf	hf	sn10	sn9	sn8	(d15~ d8)
sn7	sn6	sn5	sn4	sn3	sn2	sn1	sn0	(d7 ~ d0)

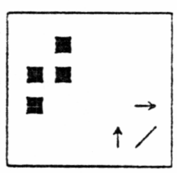
4th
HPOSITION

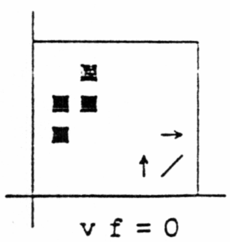
							hp8	(d15~ d8)
hp7	hp6	hp5	hp4	hp3	hp2	hp1	hp0	(d7 ~ d0)

Blank portions can be utilized freely for software.

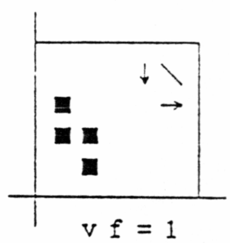
vp9～vp0	:	V POSITION
hp8～hp0	:	H POSITION
hs1, hs0	:	SPRITE's H SIZE
vs1, vs0	:	SPRITE's V SIZE
ld6～ld0	:	LinkData
pri	:	PRIORITY BIT(See PRIORITY)
cp1, cp0	:	COLOR PALETTE SSELECTION BIT(See COLOR PALETTE)
vf	:	V REVERSE BIT 1:REVERSE
hf	:	H REVERSE BIT 1:REVERSE
sn10～sn0	:	SPRITE PATTERN GENERATOR NUMBER

By using REVERSE bit vf, hf, V and H REVERSE per SPRITE is possible.

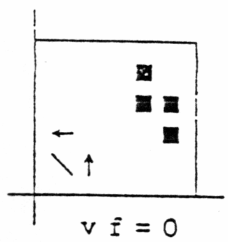




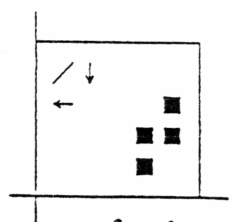
vf=0
hf=0



vf=1
hf=0



vf=0
hf=1



vf=1
hf=1

◇ SPRITE SIZE ◇

Per SPrite dot number can be set on a cell unit basis, by using vs1, vs0, hs1, hs0.

V		
vs1	vs0	Number of cell
0	0	1 (8 dots)
0	1	2 (16 dots)
1	0	3 (24 dots)
1	1	4 (32 dots)

H		
hs1	hs0	Number of cell
0	0	1 (8 dots)
0	1	2 (16 dots)
1	0	3 (24 dots)
1	1	4 (32 dots)

However, in INTERLACE MODE 2, one cell is comprised of 8×16dots, therefore, the number of V dots is two times (as compared to INTERLACE MODE 1).

◇ SPRITE's DISPLAY CAPACITY ◇

The number of SPRITE's maximum display varies depending on H resolution setting.

RESOLUTION	NO.OF DISPLAY	NO.OF DISPLAY PER LINE	DISPLAY DOT PER LINE
H 32CELL	MAX.64 SPRITES	MAX.16 SPRITES	MAX.256 DOTS
H 40CELL	MAX.80 SPRITES	MAX.20 SPRITES	MAX.320 DOTS

SPRITE is displayed in the sequential order of PRIORITY.

Example:

○With H size 1 cell, when 30 SPRITES are intended to be displayed on the same line, up to 16 SPRITES counting from the one having highesest priority(in the H 32 cell mode) and 20 SPRITEs in the H 40 cell mode can be displayed, due to the limitation of display per line.

○With H size 4 cells, when 16SPRITEs are intended to be displalyed on the same line, up to 8 SPRITEs counting from the one having the highest priority(in the H 32 cell mode) and 10 SPRITEs in the H 40 cell mode can be displayed, due to the limitation of DISPLAY dots.

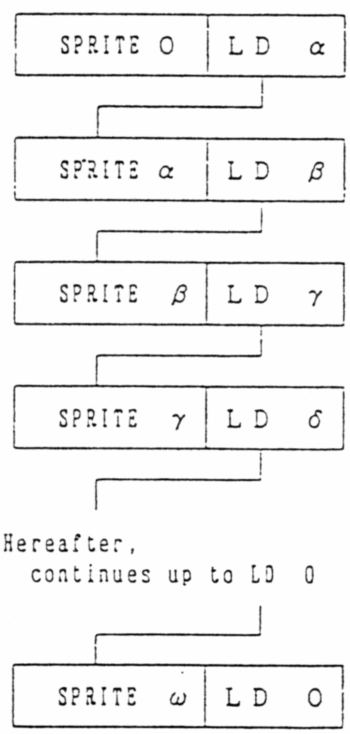
○With H size 3 cells, when 16SPRITEs are intended to be displayed in the same line, 11 SPRITEs counting from the one having the highest priority(as for 11th one, however, only for 16 dots from the left end) in the H 40 cell mode can be displayed, due to the limitation of the display dots.

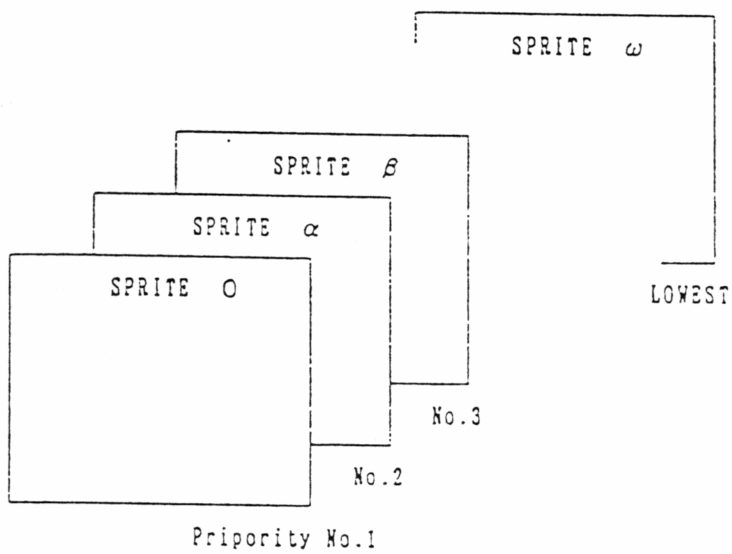
◇ PRIORITY BETWEEN SPRITES ◇

PRIORITY between SPRITEs is designated by each SPRITE's LINK DATA.
With SPRITE 0 being PRIORITY No. 1, the Sprite No. 2 written inthe LINK DATA there of will be PRIORITY No. 2. PRIORITY No. 2 SPRITE's LINK DATA shows PRIORITY No. 3 SPRITE, PRIORITY No. 3 SPRITE's LINK DATA shows PRIORITY No. 4 SPRITE. In this way, PRIORITY is sequentially designated by each SPRITE's LINK DATA and thus it is in LIST form. The value that can be set in the LINK DATA is 0～(Number of max. DISPLAY on one screen minus one). Be sure to set 0 to the lowest priority SPRITE's LINK DATA.

When 0 is given to the SPRITE LINK DATA, LIST ends at that SPRITE and the PRIORUTY will become the lowest. Even in the case that the number of SPRITEs linked to LIST is less than the max. display quantity (64 or 80), the remaining SPRITEs not linked to SPRITE will not be displayed.

When value other than those specified is set to LINK DATA, or 0 is not set to the lowest PRIORITY SPRITE LINK DATA, ordinary functioning is not quaranteed.





SETTING EXAMPLE

	LinkData
SPRITE 0	2
SPRITE 1	10
SPRITE 2	1
SPRITE 3	4
SPRITE 4	5
SPRITE 5	15
SPRITE 6	----
SPRITE 7	0
SPRITE 8	----
SPRITE 9	----
SPRITE 10	11
SPRITE 11	13
SPRITE 12	----
SPRITE 13	3
SPRITE 14	----
SPRITE 15	7
SPRITE 16	----

表示の優先順位
SPRITE 0
SPRITE 2
SPRITE 1
SPRITE 10
SPRITE 11
SPRITE 13
SPRITE 3
SPRITE 4
SPRITE 5
SPRITE 15
SPRITE 7

The 11SPRITEs shown in the DISPLAY PRIORITY are displayed on the screen. SPRITE No. 6, 8, 9, 12, 14, and 16 onward are not displayed becouse they are not linked with LINK DATA LIST.

◇ SPRITE PATTERN GENERATOR ◇

The SPRITE PATTERN GENERATOR with VRAM 0000H as BASE ADDRESS, expresses one pattern on a basis of 8×8 dots. 32 bytes are required to define one pattern. Every 32 bytes, one pattern is expressed in the sequence of PATTERN GENERATOR 0, 1, 2 · · ·. The relationship of DISPLAY PATTERN and MEMORY is the same as in PATTERN GENERATOR. Also, SPRITE SIZE and PATTERN GENERATOR relationship is as follows:

V 1 cell
H 1 cell

0

V 1 cell
H 2 cell

0	1

V 1 cell
H 3 cell

0	1	2

V 1 cell
H 4 cell

0	1	2	3

V 2 cell
H 1 cell

0
1

V 2 cell
H 2 cell

0	2
1	3

V 2 cell
H 3 cell

0	2	4
1	3	5

V 2 cell
H 4 cell

0	2	4	6
1	3	5	7

V 3 cell
H 1 cell

0
1
2

V 3 cell
H 2 cell

0	3
1	4
2	5

V 3 cell
H 3 cell

0	3	6
1	4	7
2	5	8

V 3 cell
H 4 cell

0	3	6	9
1	4	7	A
2	5	8	B

V 4 cell
H 1 cell

0
1
2
3

V 4 cell
H 2 cell

0	4
1	5
2	6
3	7

V 4 cell
H 3 cell

0	4	8
1	5	9
2	6	A
3	7	B

V 4 cell
H 4 cell

0	4	8	C
1	5	9	D
2	6	A	E
3	7	B	F

§11 PRIORITY

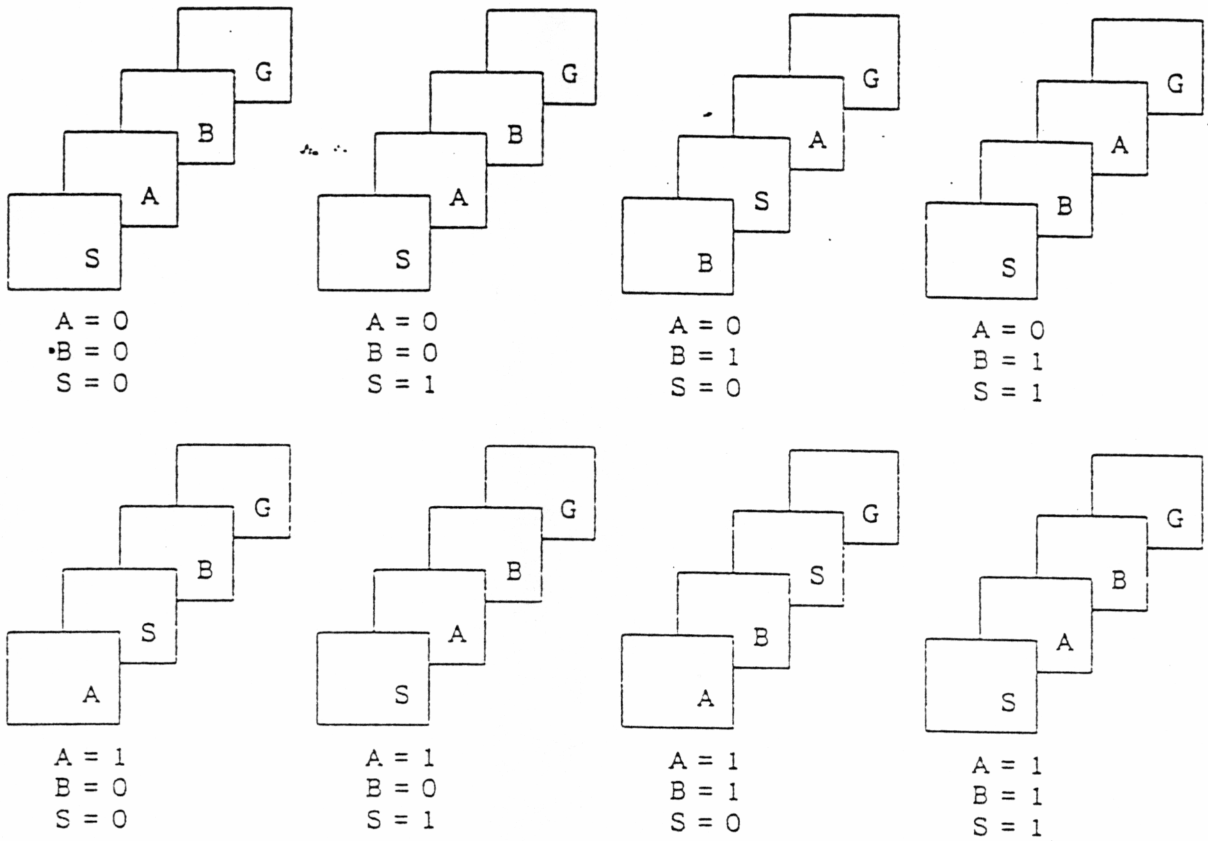
PRIORITY between SPRITE, SCROLL A and SCROLL B can be designated.
PRIORITY can be designated by each PATTERN NAME and ATTRIBUTE PRIORITY bit.. It will be set for the SCROLL SCREEN on a cell unit basis and for each SPRITE. By combining each priority bit, PRIORITY will be as follows:
However, the BACKGRAOUND PRIORITY is always the lowest.

S pri	A pri	B pri	PRIORITY
0	0	0	S>A>B>G
1	0	0	S>A>B>G
0	1	0	A>S>B>G
1	1	0	S>A>B>G
0	0	1	B>S>A>G
1	0	1	S>B>A>G
0	1	1	A>B>S>G
1	1	1	S>A>B>G

S : SPRITE
A : SCROLL A
B : SCROLL B
G : BACKGROUND

Also, by combining S/TEN (REG. #12) and the above priority, SHADOW-HIGHLIGH effect function can be utilized.

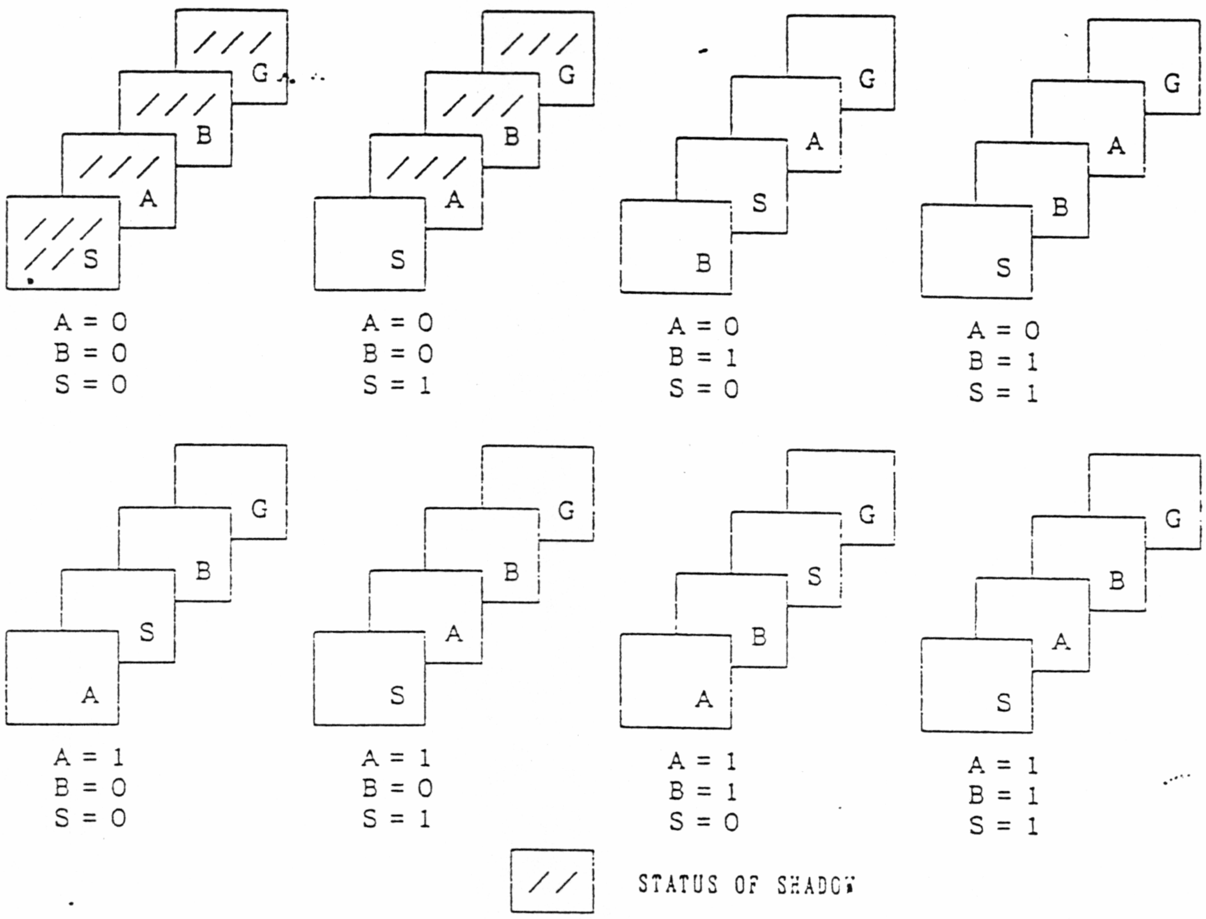
S/TEN=0



The above shows PRIORITY situation of SPRITE, SCROLL A, SCROLL B and BACKGROUND. The dot to which COLOR CODE 0 is designated is transparent, therefore, either one of SCROLL SCREEN A, B, or BACKGROUND, the priority of which is one step lower than the transparent one, will appear.

S/TEN=1

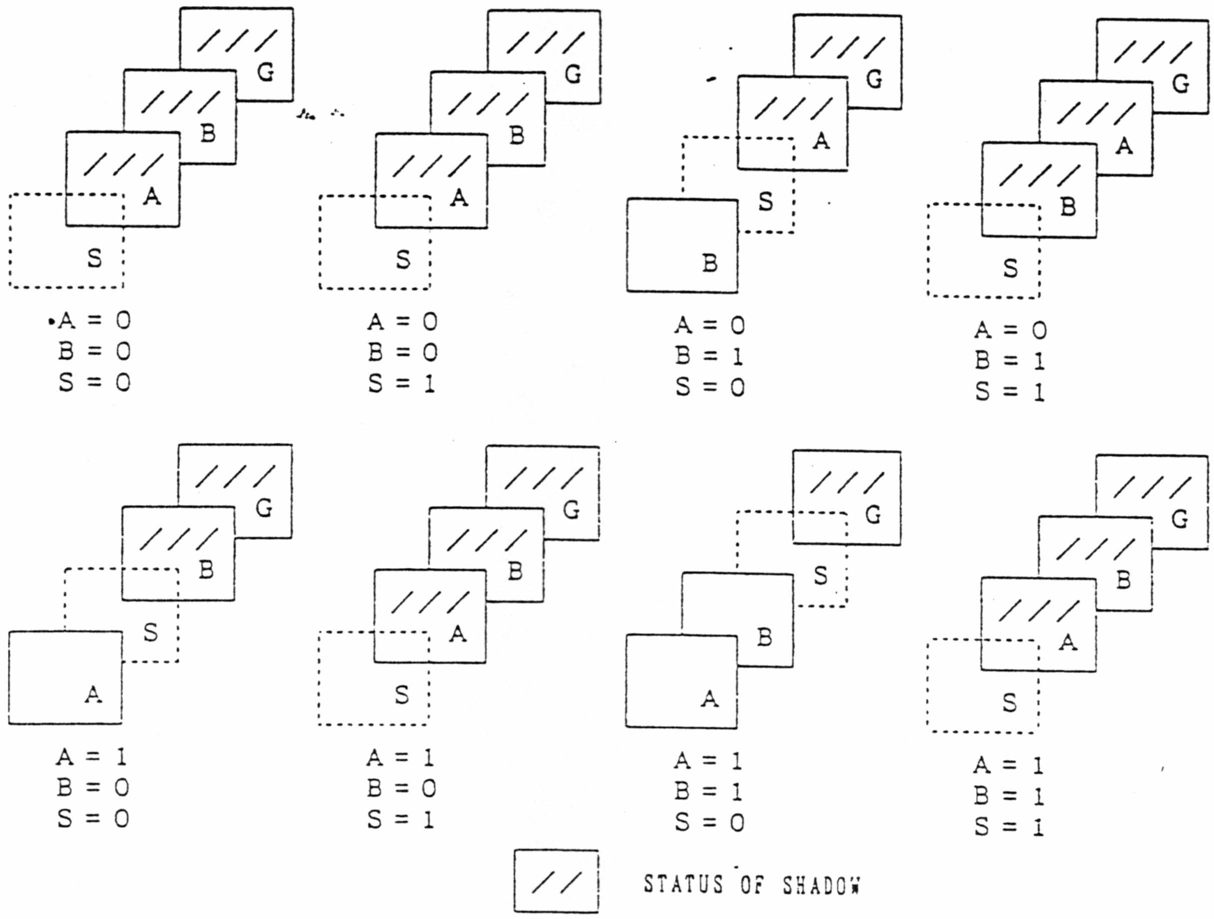
SPRITE COLOR PALETTE 0 ~ 3 , COLOR CODE 0 ~ 15
COLOR PALETTE 3 , COLOR CODE 0 ~ 13



Where S/TEN=1, when the PRIORITY bit of both SCROLL A and SCROLL B is 0, there will be SHADOW. For the color status, refer to the color palette.

S/TEN = 1

SPRITE COLOR PALETTE 3 , COLOR CODE 15

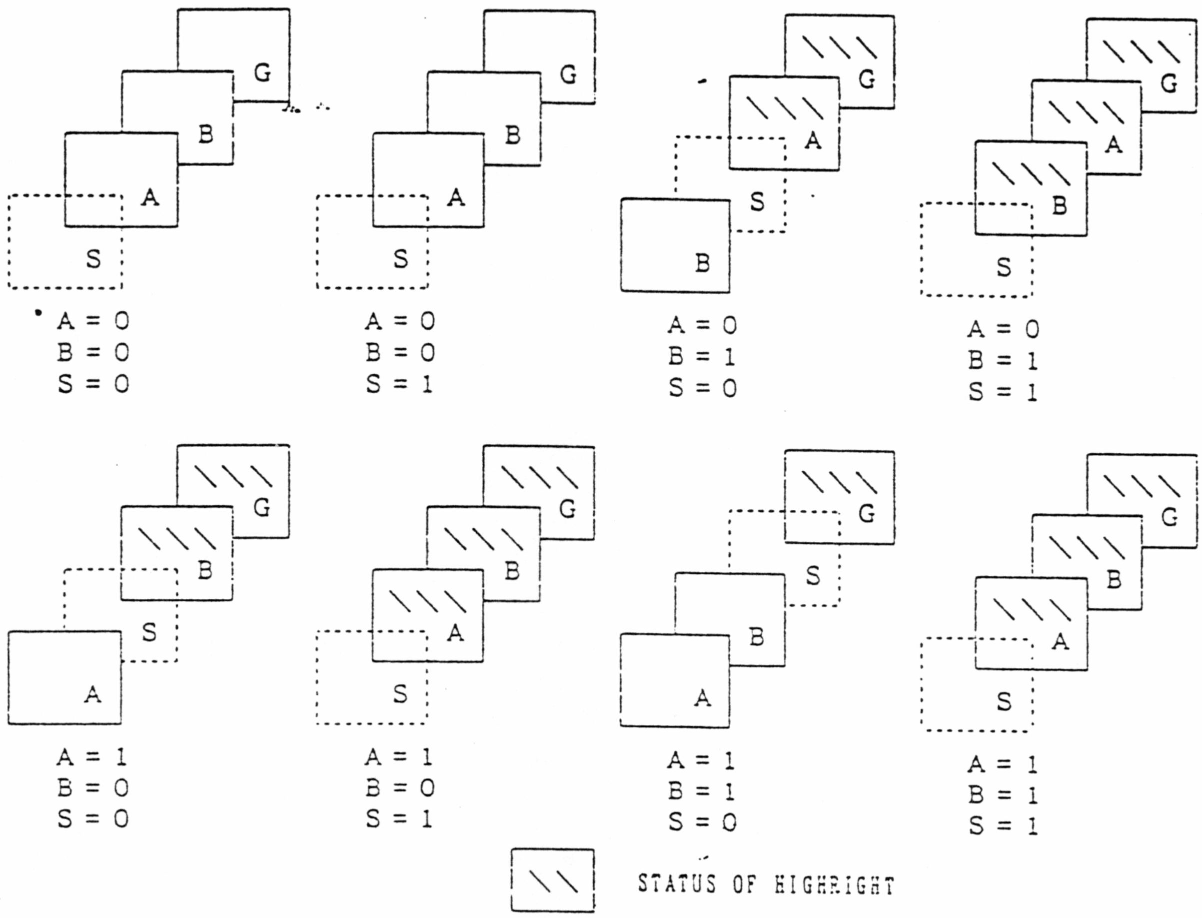


The dots for the SPRITE COLOR code 15 work as a SHADOW operator on the screen, the PRIORITY of which is lower than the SPRITE.

Since SPRITE dot works as an operator, this will not be displayed.

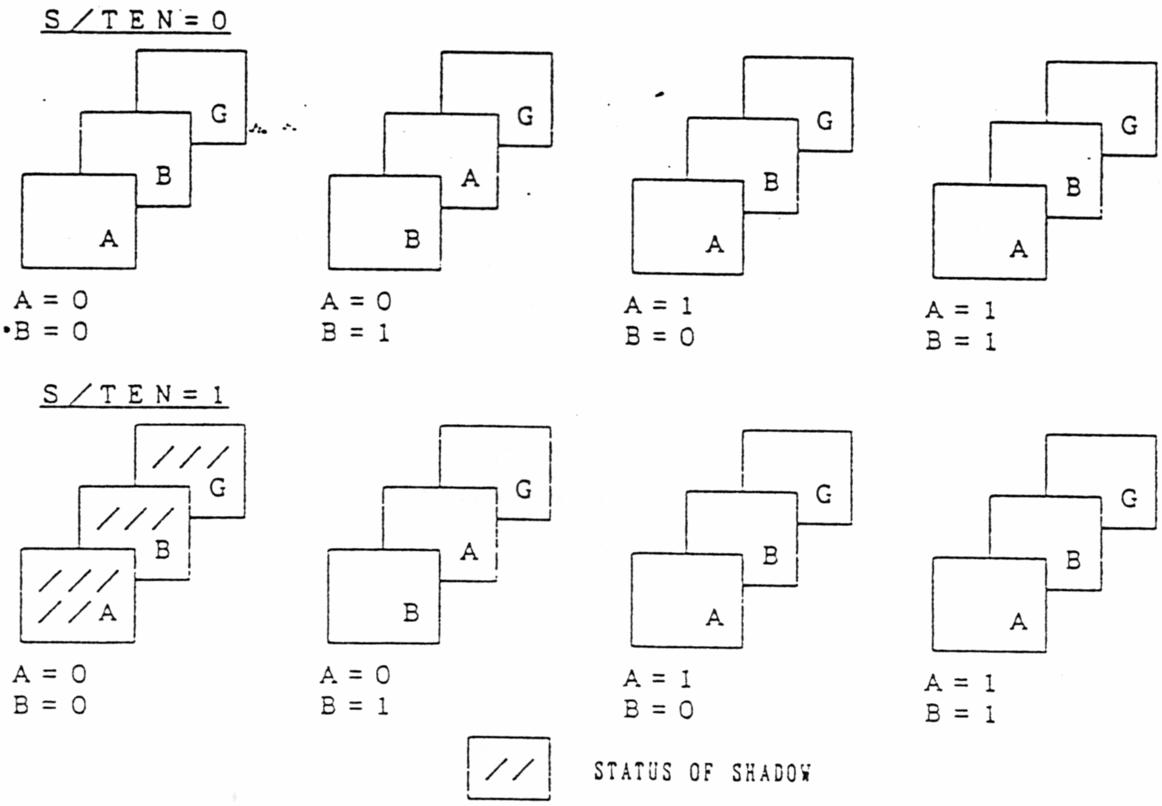
S/TEN=1

SPRITE COLOR PALETTE 3 , COLOR CODE 14



The dots of SPRITE COLOR CODE 15 work as an operator on the screen, the priority of which is lower than SPRITE.
Since SPRITE dots work as an operator, this will not be displayed.

When SPRITE is not related to PRIORITY, the following PRIORITY applies.



§12 COLOR PALETTE

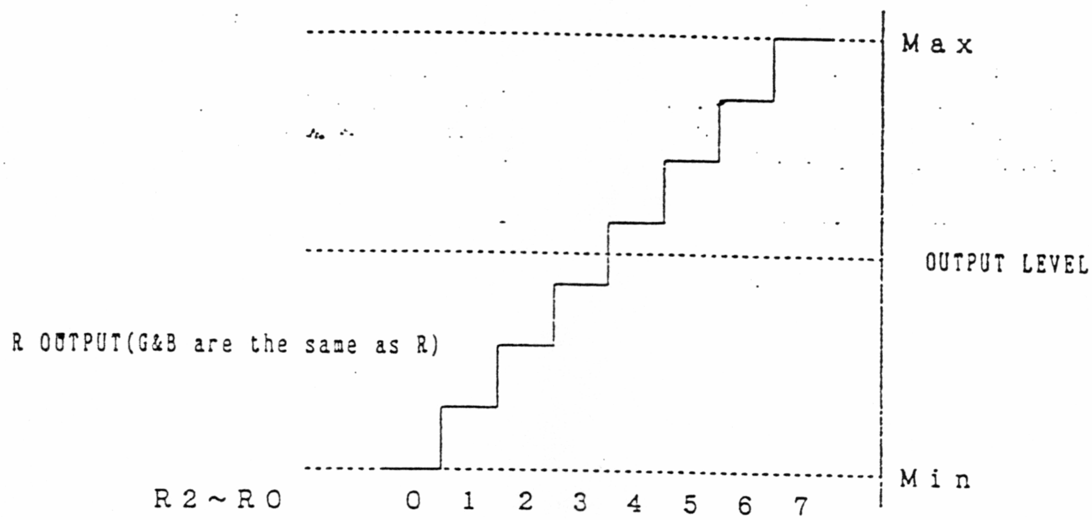
One dot is comprised of 4 bits and can desifgnate the 0~15 colors. Also, 0~3 color palette can be designated by SCROLL screen on a cell basis and by each SPRITE. CRAM data are as follows. Since each of R, G, and B has 3 bits, colors can be freely selected out of 512 colors.

DATA									
	0	0	0	0	B2	B1	B0	0	(D15~ D8)
	G2	G1	G0	0	R2	R1	R0	0	(D7 ~ D0)

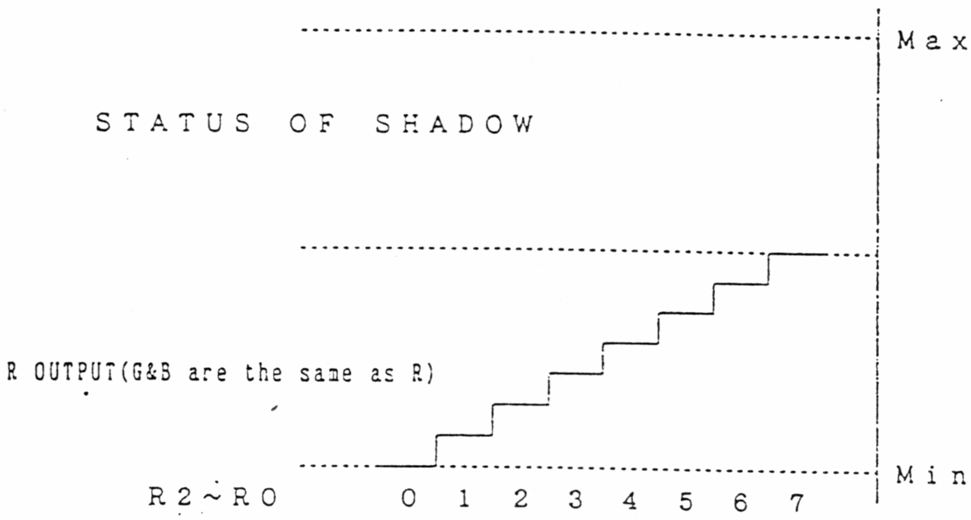
The relationships between CRAM address, palette and color code are follows:
However, in the case of each palette's color code 0, the color for the SCROLL A, SCROLL B, WINDOW and SPRITE is SEE THROUGH irrespective of RBG designation.

ADDRESS	BLUE	GREEN	RED	PALETTE	CODE	REMARKS
00H 02H 04H : : : 1AH 1CH 1EH				0	0 1 2 : : : 13 14 15	The 0~15 colors designated by rgb will be displayed.
20H : 3EH				1	0 : 15	Same as PALETTE 0
40H : 5EH				2	0 : 15	Same as PALETTE 0
60H 62H : 7AH 7CH 7EH				3	0 1 : 13 14 15	Same as PALETTE 0 ※For 14 and 15, refer to PRIORITY

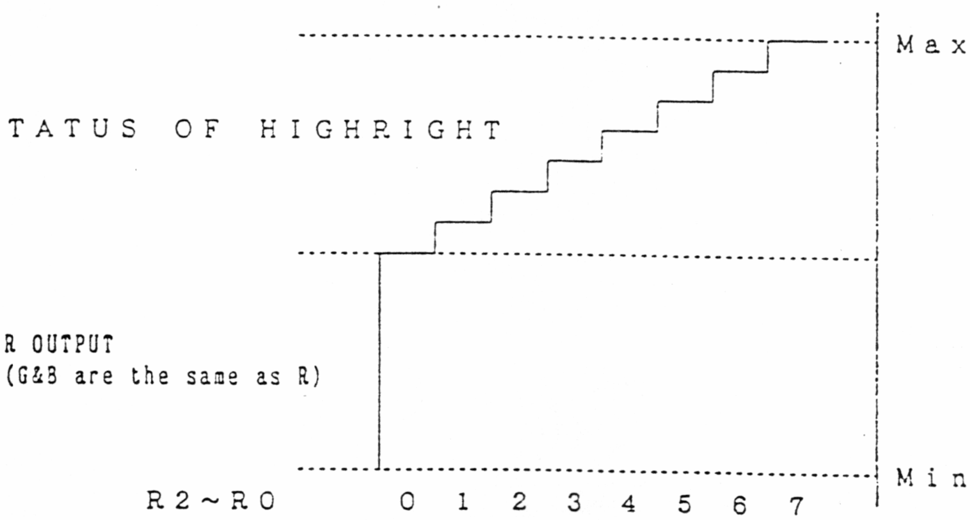
RGB bit and display are as follows:



STATUS OF SHADOW



STATUS OF HIGHRIGHT



§13 INTERLACE MODE

RASTER SCAN MODE can be changed by setting LSM0 and LSM1 (RGB. #12).

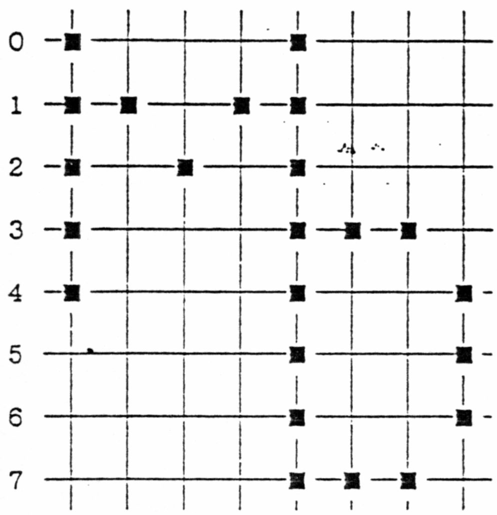
LSM1	LSM0	RASTER SCAN MODE
0	0	NONINTERLACE MODE
0	1	In the NON-INTERLACE mode, the same PATTERN is displayed on the rasters of even and odd numbered fields. (INTERLACE 1)
1	1	In the INTERLACE MODE, the different PATERN is displayed on the rasters of even and odd numbered fields. (INTERLACE 2)

In the INTERLACE MODE and INTERLACE 1, one cell is defined by 8×8 dots and in INTERLACE 2, 8×16 dots. For DISPLAY, one cell consists of 8×8 dots in the NONINTERLACE MODE and in the INTERLACE MODE, 8×16 dots.

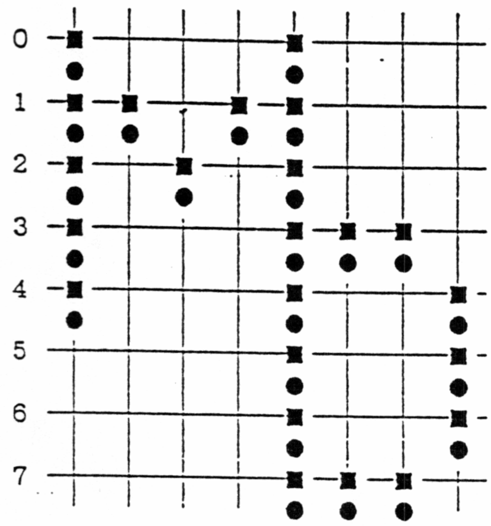
In any case, number of cells in one screen are the same.

Depending on the type of DISPLAY, in the case of INTERLACE DISPLAY, there may occur a serious blur in the vertical direction, therefore, when using the DISPLAY, pay careful attention in this regard.

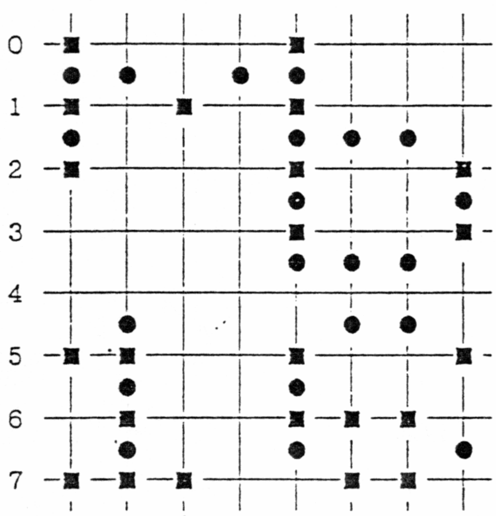
NON-INTERLACE



INTERLACE 1



INTERLACE 2



■ FIELD NO. 1

● FIELD NO. 2

3. BACKWARD COMPATIBILITY MODE

In the case of BACKWARD COMPATIBILITY MODE, the MEGA DRIVE differs from the original MarkIII&MASTER SYSTEM in the following points:

◇ MARKIII (MS-JAPAN) ◇

○ OS-ROM is not incorporated.

ROM CARTRIDGE/CARD selections are made by hardware in the same manner as in the case of MARKIII.
START UP SLOT number is not written in 0C000H.
START UP Sega logo is not displayed.

○ FM sound source is not incorporated.

FM sound is incorporated in MS-JAPAN (standard) and MARKIII (optional) (OPLL), however, MEGA DRIVE has no option for that, although connection is possible.
Consider the MEGA DRIVE's Japanese Specifications as that of MARKIII with MS-JAPAN's JOYSTICK Port, or as MS-JAPAN without FM sound source and OS-ROM.

◇ MASTER SYSTEM ◇

○ OS-ROM is not incorporated.

0C000H~0DFFFH RAM is not clear on POWER-UP.
RAM 0C000 has no meaningful value.
START UP Sega logo not displayed.

○ FM sound source is not incorporated.

FM sound source is incorporated in MS by option (OPLL). However, MEGA DRIVE has no option, although connection is possible.

Please regard the MEGA DRIVE overseas version as a MASTER SYSTEM without an Operating System ROM.

◇ RAM BOARD ◇

In the MEGA DRIVE's MARKIII&MASTER SYSTEM BACKWARD COMPATIBILITY MODE, the RAM BOARD for development (for which D-RAM was used) can not be used due to the problem of REFRESH. The other BOARDs for development (which utilizes S-RAM) can be used without any problem.

4. SYSTEM I/O

MEGA DRIVE SYSTEM I/O area assignment starts from $A00000, with the Z80 SUB-CPU's memory area.

§1 VERSION NO.

Indicates the Mega Drive's hardware version.

$A10001	MODE	VMOD	DISK	RSV	VER3	VER2	VER1	VER0

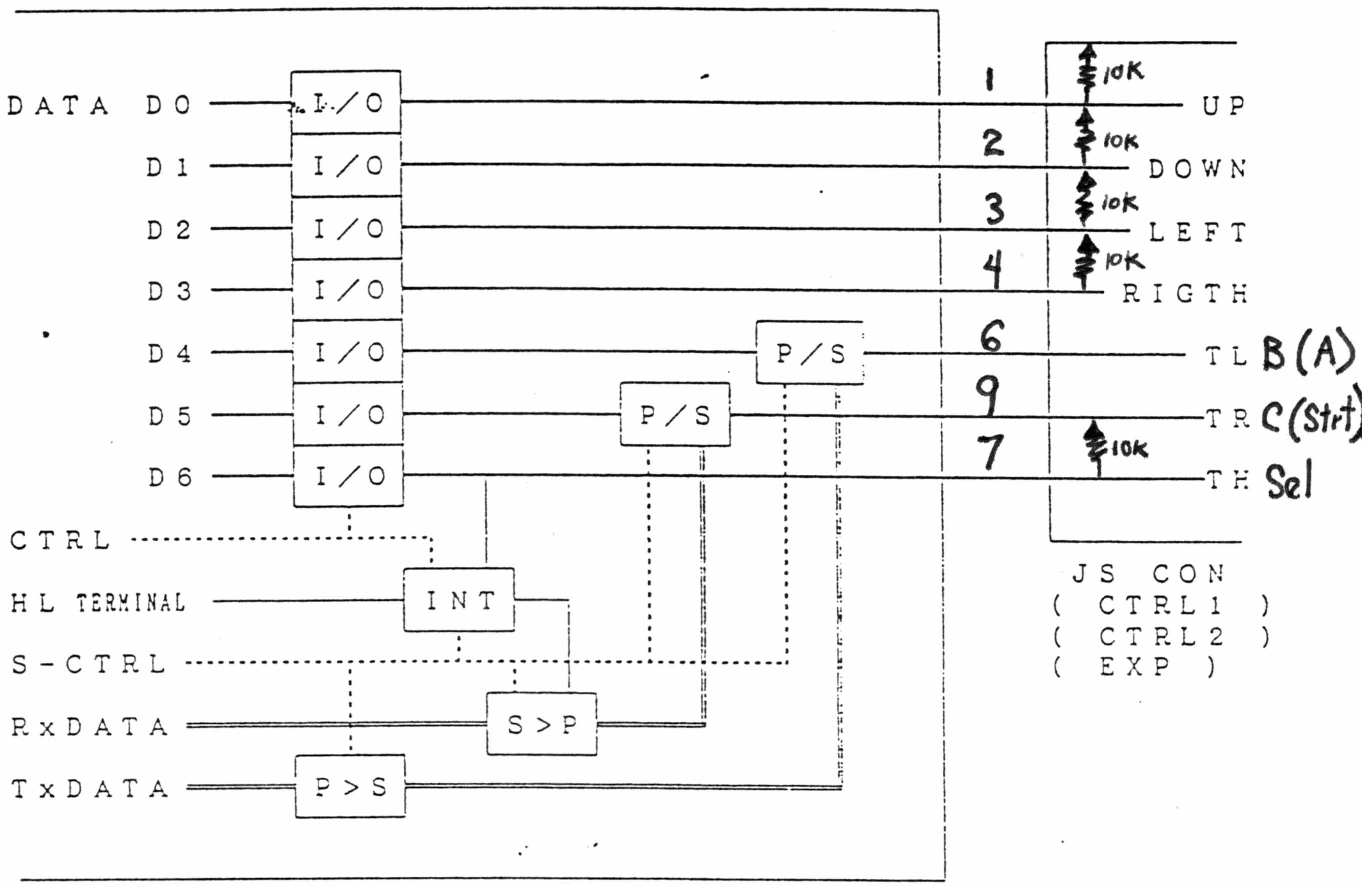
MODE	(R)	0: Domestic Model 1: Overseas Model
VMOD	(R)	0: NTSC CPU clock 7.67MHZ 1: PAL CPU clock 7.60MHZ
DISK	(R)	0: FDD unit connected 1: FDD unit not connected
RSV	(R)	Currently not used
VER3~0	(R)	MEGA DRIVE version is indicated by $0~$F. The present hardware version is indicated by $0.

§2 I/O PORT

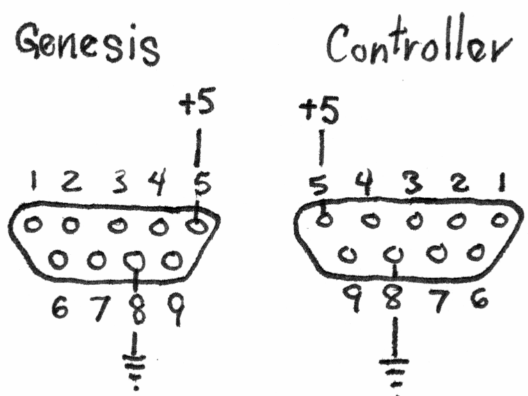
The MEGA DRIVE has the three general purpose I/O ports, CTRL1, CTRL2 and EXP. Although each port differs from the others in physical shape, it functions in the same manner. Each port has the following 5 REGISTERs for CONTROL.

DATA	(PARALLEL DATA)	:	R/W
CTRL	(PARALLEL CONTROL)	:	R/W
S-CTRL	(SERIAL CONTROL)	:	R/W
TxDATA	(Txd DATA)	:	R/W
RxDATA	(Rxd DATA)	:	R

The relationship between REGISTERs in as follows:



I/O	:	I/O change
P/S	:	PARALLEL/SERIAL MODE change
INT	:	INTERRUPT CONTROL
S>P	:	SERIAL-PARALLEL CONVERSION
P>S	:	PARALLEL-SERIAL CONVERSION



Mapping is as follows.

```
$A10003 : DATA 1  ( CTRL1 )
$A10005 : DATA 2  ( CTRL2 )
$A10007 : DATA 3  ( EXP )
$A10009 : CTRL 1
$A1000B : CTRL 2
$A1000D : CTRL 3
$A1000F : TxDATA 1
$A10011 : RxDATA 1
$A10013 : S-CTRL 1
$A10015 : TxDATA 2
$A10017 : RxDATA 2
$A10019 : S-CTRL 2
$A1001B : TxDATA 3
$A1001D : RxDATA 3
$A1001F : S-CTRL 3
```

Both BYTE and WORD access are possible. However, in the case of WORD access, only the lower byte is meaningful.

DATA shows the status of each port. The I/O direction of each bit is set by CTRL and S-CTRL.

DATA	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0

```
PD7  (RW)
PD6  (RW)  TH
PD5  (RW)  TR
PD4  (RW)  TL
PD3  (RW)  RIGHT
PD2  (RW)  LEFT
PD1  (RW)  DOWN
PD0  (RW)  UP
```

CTRL designates the I/O direction ofeach port and the INTERRUPT CONTROL of TH.

CTRL

INT	PC6	PC5	PC4	PC3	PC2	PC1	PC0

INT	(RW)	0: TH-INT	PROHIBITED
		1: //	ALLOWED
PC6	(RW)	0: PD6	INPUT MODE
		1:	OUTPUT MODE
PC5	(RW)	0: PD5	INPUT MODE
		1:	OUTPUT MODE
PC4	(RW)	0: PD4	INPUT MODE
		1:	OUTPUT MODE
PC3	(RW)	0: PD3	INPUT MODE
		1:	OUTPUT MODE
PC2	(RW)	0: PD2	INPUT MODE
		1:	OUTPUT MODE
PC1	(RW)	0: PD1	INPUT MODE
		1:	OUTPUT MODE
PC0	(RW)	0: PD0	INPUT MODE
		1:	OUTPUT MODE

S-CTRL is for the status, etc. of each port's mode change, baud rate and SERIAL.

S-CTRL

BPS1	BPS0	SIN	SOUT	RINT	RERR	RRDY	TFUL

SIN	(RW)	0: TR-PARALLEL MODE	
		1: SERIAL IN	
SOUT	(RW)	0: TL-PARALLEL MODE	
		1: SERIAL OUT	
RINT	(RW)	0: RxdREADY	INTERRUPT PROHIBITED
		1: //	INTERRUPT ALLOWED
RERR	(R)	0:	
		1: RxdERROR	
RRDY	(R)	0:	
		1: RxdREADY	
TFUL	(R)	0:	
		1: TxdFULL	

BPS1	BPS0	bps
0	0	4800
0	1	2400
1	0	1200
1	1	300

§3 MEMORY MODE

The MEGA DRIVE is able to generate internally the REFRESH signal for the D-RAM development cartridge. When using the development cartridge set to D-RAM MODE. In the case of a production cartridge, set to ROM MODE.

Only D8 of address $A11000 is effective and for WRITE ONLY.

```
$A11000     D8( W)  0:ROM MODE
                    1:D-RAM MODE
```

ACCESS to $A11000 can be based on BYTE.

§4 Z80 CONTROL

◇ Z80 BUSREQ ◇

When accessing the Z80 memory from the 68000, first stop the Z80 by using BUSREQ. At the time of POWER ON RESET, the 68000 has access to the Z80 bus.

```
$A11100     D8( W)  0: BUSREQ CANCEL
                    1: BUSREQ REQUEST
              (R )  0: CPU FUNCTION STOP,ACCESSIBLE
                    1: FUNCTIONING
```

Access to Z80 AREA in the following manner.

(1) Write $0100 in $A11100 by using a WORD access.
(2) Check to see that D8 of $A11100 becomes 0.
(3) Access to Z80 AREA.
(4) Write $0000 in $A11100 by using a WORD access.

Access to $A11100 can also be based on BYTE.

◇ Z80 RESET ◇

The 68000 may also reset the Z80.
The Z80 is automatically reset during the MEGADRIVE hardware's POWER ON RESET sequence.

```
$A11200     D8( W)  0:RESET REQUEST
                    1:RESET CANCEL
```

Access to $A11100 can also be based on BYTE.

§5 Z80 AREA

Mapping is performed starting from $A00000 for Z80, a SUB-CPU.
As viewed from 68000, the memory map will be as follows:

Address	Area
$A00000	SOUND RAM
$A02000	PROHIBITED
$A04000	SOUND CHIP
$A04004	PROHIBITED
$A06000	BANK REGISTER
$A06002	PROHIBITED
$A08000	PROHIBITED

◇ SOUND RAM ◇

This is for the z80 program.
Access from 68000 by BYTE.

◇ SOUND CHIP ◇

This is the mapping area for FM sound source (YM2612). When accessing from 68000, use BYTE due to timing problem.

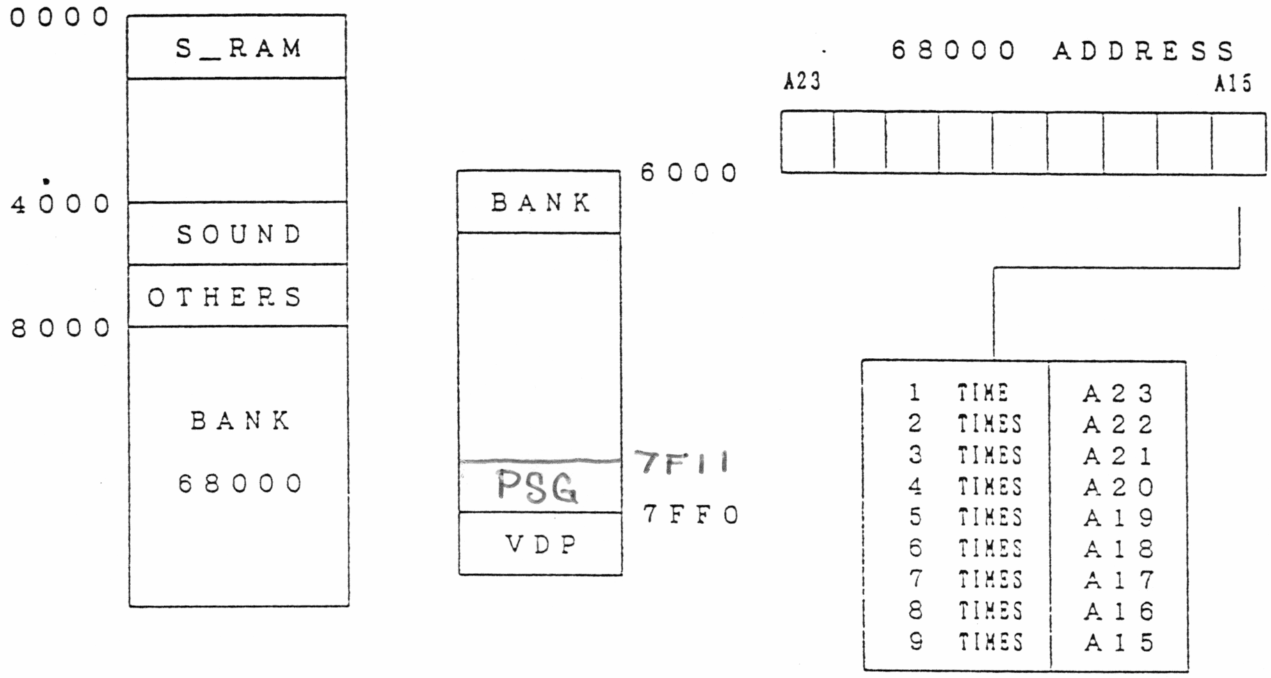
◇ BANK REGISTER ◇

Access to the 68000 side MEMORY AREA from Z80 will be based on a 32K BYTE unit. At this time, this REGISTER sets which BANK is to be accessed. ~~Registering from 68000 can be set, however,~~ do not access to Z80 BAnk MEMORY AREA by 68000.

GTB#13 - Only Z80 can set this register.

SETTING METHOD

When accessing to the 68000 side addresses from Z80 side, all the addresses can be classified into BANKs. BANK can be set by writing 9 times in 0 bit of 6000 (Z80 address). The 9 bits correspond to 68000 address 15~23 as shown below:



5. VRAM MAPPING

In VRAM, there are various TABLEs and PATTERN GENERATORs as stated below. Among those, the base address of PATTERN GENETRATOR TABLE and SPRITE GENERATOR TABLE are 0000H and fixed. However, the other base addresses can be freely assigned in VRAM by setting VDP REGISTER. Also, AREA can be overlapped, therefore, TABLE can be commonly used by SCROLL screen and WINDOW, for example.

- SCROLL A PATTERN NAME TABLE Max. 8K Byte.
  Base address designated by REGISTER #2.
- SCROLL B PATTERN NAME TABLE Max. 8K Byte.
  Base address designated by REGISTER #4.
- WINDOW PATTERN NAME TABLE Varies by H Resolution
  Base address designated by REGISTER #3.
- H SCROLL DATA TABLE 1K Byte
  Base address designated by REGISTER #13.
- SPRITE ATTRIBUTE TABLE Varies by H Resolution
  Base address designated by REGISTER #5.
- PATTERN GENERATOR TABLE
  Base address is 0000H (fixed).
- SPRITE GENERATOR TABLE
  Base address is 0000H (fixed).

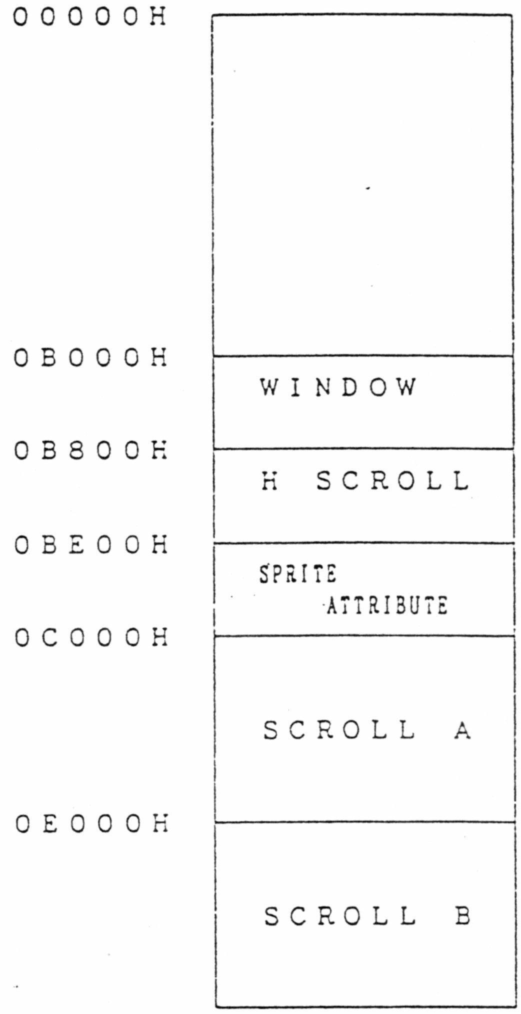
There are 1K Bytes for H SCROLL TABLE, however, as for display 896 Bytes in V28 cell mode and 960 bytes in V30 cell mode. There are 2K bytes for WINDOW PATTERN NAMETABLE in H32 cell mode, and 4K byte area in H40 cell mode. For details refer to WINDOW. There are 512 bytes for SPRITE ATTRIBUTE TABLE in H32 cell and 1K byte area in H40 cell mode. However, as for display, there are 640 bytes in H40 cell mode.

Setting exsample

1 H32 cell mode

- ○ SCROLL A PATTERN NAME TABLE
  8K Bytes from 0C000H : REG. #2=$30
- ○ SCROLL B PATTERN NAME TABLE
  8K Bytes from 0E000H : REG. #4=$07
- ○ WINDOW PATTERN NAME TABLE
  2K Bytes from 0B000H : REG. #3=$2C
- ○ H SCROLL DATA TABLE
  1K Bytes from 0B800H : REG. #13=$2E
- ○ SPRITE ATTRIBUTE TABLE
  512 Bytes from 0BE00H : REG. #5=$5F

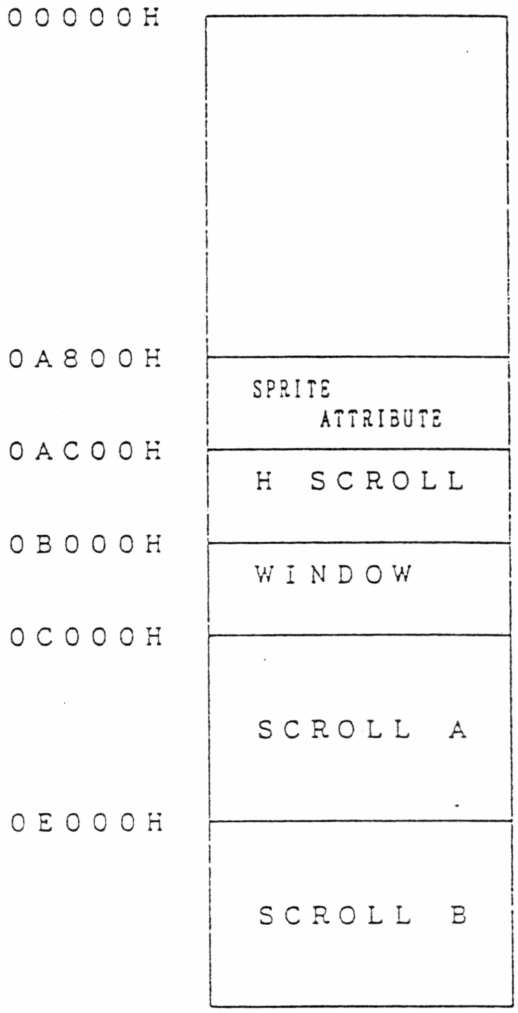
Unoccupied area is used as PATTERN GENERATOR and SPRITE GENERATOR.



2 H40 cell mode

- O SCROLL A PATTERN NAME TABLE
  8K Bytes from 0C000H : REG. #2=$30
- O SCROLL B PATTERN NAME TABLE
  8K Bytes from 0E000H : REG. #4=$07
- O WINDOW PATTERN NAME TABLE
  4K Bytes from 0B000H : REG. #3=$2C
- O H SCROLL DATA TABLE
  2K Bytes from 0AC00H : REG. #13=$2B
- O SPRITE ATTRIBUTE TABLE
  1K Bytes from 0A800H : REG. #5=$54

Unoccupied area is used as PATTERN GENERATOR and SPRITE GENERATOR.



PRECAUTIONS FOR M5 SOFTWARE PROGRAMMING

When programming the M5 software, pay attention to the following:

I The program od DMA (RAM, ROM→VRAM, CRAM, VSRAM) should be resident in RAM, or it should be as in LIST1 for example. However, in either one on the above 2 cases, a long word access is not possible as regards the last VRAM address set.

II I.D should be as in the next page.

III Put LIST2 at your program's start. This is the U. S security software.

LIST1

```
DMA_RAM:
        lea     vdp_cmd,An              ; vdp_cmd = $c00000
                                        ; An = ADDRESS REGISTER
        ;  Set source ADDRESS to VDP REGISTER
        ;  Set DATA LENGTH to VDP REGISTER
        move.l  xx,ram0                 ; xx: DISTINATION ADDRESS
                                        ; ram0 :WORK RAM
        move.w  ram0,(An)
        move.w  ram0+2,(An)             ; Pay careful attention to the sequential order of 1st
                                          word and 2nd word.
        rts                             ; DESTINATION ADDRESS should be set by WORD and not by
                                          LONG WORD.
```

LIST2

```
        move.b  $a10001,d0              ;  Get version number
        andi.b  #$0f,d0                 ;
        beq.b   ?0                      ;  If not version #0
        move.l  #'SEGA',$a14000         ;  Output ASCII
?0:
```

ROM CARTRIDGE DATA FOR MEGA DRIVE

Write in ROM's 100H~1FFH.

100H:	'SEGA MEGA DRIVE '	1
110H:	'(C)SEGA 1988.JUL'	2
120H:	GAME NAME(DOMESTIC)	3
150H:	GAME NAME(OVERSEAS)	4
180H:	'GM XXXXXXXX-XX'	5
18EH:	$XXXX	6
190H:	CONTROL DATA	7
1A0H:	$000000,$XXXXXX	8
1A8H:	$FF0000,$FFFFFF	9
1B0H:	EXTERNAL RAM DATA	10
1BCH:	MODEM DATA	11
1C8H:	MEMO	12
1F0H:	Country in which the product can be released.	13

1: SEGA,system name and TITLE in common with all ROMs.
2: Copyright notice and year/month of release (Firm name in ASCII 4 character.)
3: Game name for Domestic(JIS KANJI CODE OK)
4: Game name for overseas market(JIS KANJI CODE OK)
5: Type of CARTRIDGe and Products, NO., Version No.

TYPE GAME: GM
EDUCATION: AI
NO. PRODUCT NO.
VER. Data varies depending on the type of ROm or software version.

6: Check sum
7: I/O use support data

JOYSTICK FOR MS	: 0
JOYSTICK	: J
KEYBOARD	: K
SERIAL(RS232C)	: R
PRINTER	: P
TABLET	: T
CONTROL BALL	: B
PADDLE CONTROLLER	: V
FDD	: F
CDROM	: C

8: ROM capacity START ADDRESS,END ADDRESS
9: RAM capacity START ADDRESS,END ADDRESS
10: When no external RAM is mounted,fill the address by a space code,and when it is mounted,follow the following:

```
1B0H:  dc.b    'RA',%1x1yz000,%00100000
   x      1 for BACKUP and 0 if not BACKUP
   yz     10 if even address only,  11 if odd address only
          00 if both even and odd address
1B4H:  dc.l    RAM start address,RAM end address
```

11: If corresponding to MODEM,fill it by space code and if not,follow the following.

```
1BCH:  dc.b    'MO','xxxx','yy.z'
       xxxx   Firm name,the same as in 2
       yy     MODEM NO.
       z      Version
```

13: Data of the countries in which the product can be released.

JAPAN	: J
USA	: U
EUROPE	: E

Be sure to input a space code in the unoccupied 1~7, 9~13 space.

HOW TO OBTAIN CHECK SUM

The CHECK SUM obtaining program is shown as follows. The program starts wiith OFF8000H, RAM space.
First, fill game capaciity by -1 (OFFH) and then load all of the programs. Next, load the CHECK SUM program and run the program from OFF8000H. After a while, stop running the program.
At this time, the lower WORD of DATA REGISTER 0 (d0) is the CHECK SUM value. Note that BREAK in MEMORY should be cancelled in advance.
Also, when burnins to ROM, first fill the game capacity by -1 (OFFH).

```
end_addr        equ     $1a4
        org     -$8000

start:
                move.l  (a0),d1
                addq.l  #$1,d1
                movea.l #$200,a0
                sub.l   a0,d1
                asr.l   #1,d1                   ; counter
                move    d1,d2
                subq.w  #$1,d2
                swap    d1
                moveq   #$0,d0
        ?12:
                add     (a0)+,d0
                dbra    d2,?12
                dbra    d1,?12
                nop
                nop
                nop
                nop
                nop
                nop
                nop
        ?1e:
                nop
                nop
                bra.b   ?1e
```

MAMORY MAPPING FOR EMULATION

·For the 68000 EMULATION

All address should be disabled initially : 0 to 0FFFFFF

Required areas should then be enabled as follows;

1. Program and Data are in 0 to 007FFFF
2. S-RAM is for Z-80 in 0A00000 to 0A01FFF
3. FM sound chip interface is in 0A04000 to 0A04FFF
4. I/O and Z-80 control port are in 0A10000 to 0A11FFF
5. VDP and sound control port are in 0C00000 to 0C00FFF
6. Scratch RAM is in 0FF0000 to 0FFFFFF

RAM CARD (No. 171-5642-02)

This board has two memory areas;

MAIN MAMORY (D-RAM) $000000 - $0FFFFF
BACK UP MEMORY (S-RAM) $200000 - $203FFF

1.INITIALIZE
Write 0100H into $0A11000
Write 1 into $0A130F0
(Green LED light up)

2.WRITE PROTECT
Write 3 into $0A130F0
(Red LED light up)

3.READ/WRITE
Write 1 into $0A130F0
(Red LED turns off)

4.NOTE - Emulator access to these ports should be enabled before the writes, then disabled after words.

MEGA DRIVE REGISTER'S FIXED BITS

(40cell and NTSC mode)

R0	0	0	0		0	1	0	0
1	0				0	1	0	0
2	0	0				0	0	0
3	0	0						0
4	0	0	0	0	0			
5	0							
6	0	0	0	0	0	0	0	0
7	0	0						
8	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0
10								
11	0	0	0	0	0			
12	1	0	0	0				1
13	0	0						
14	0	0	0	0	0	0	0	0

15									
16	O	O			O	O			
17		O	O						
18		O	O						
19									
20									
21									☆
22									
23									

☆ DMA cannot be porformed emulated ROM or RAM on most ICEs.

GENESIS SOUND SOFTWARE MANUAL

INDEX

This manual explains memory mapping and way of accessing especially. FM sound generation and PSG are explained another manual.

3)I/O

4000H FM1 register select (Channel 1-3)

4001H FM1 DATA

4002H FM2 register select (Channel 4-6)

4003H FM2 DATA

PSG address is in 7F11H.

② INTERRUPT

Z-80 gets the only VIDEO vertical interrupt.

This interrupt is generated 16ms period and 64ms length.

II. 68K CONTROL OF Z-80

① Z80 START UP

Z-80 OPERATION SEQUENCE.

(1) BUS REQ ON
(2) BUS RESET OFF
(3) 68K copies program into Z-80 S-RAM
(4) BUS RESET ON
(5) BUS REQ OFF
(6) BUS RESET OFF

BUS REQUEST

· BUS REQ ON

DATA 100H (WORD) → \$A11100

· BUS REQ OFF

DATA 0H (WORD) → \$A11100

RESET Z-80

- RESET ON

  DATA 0H (WORD) → $A11200

- RESET OFF

  DATA 100H (WORD) → $A11200

This period requires 26ms.
Also FM sound source is cleard at the same time.

CONFIRMATION OF BUS STATUS

This information is in $A11100 bit 0.
0 - Z-80 is using
1 - 68K can access

② Z80 HANDSHAKE

If you access the HANDSHAKE area (A00000 - A07FFF) you must use BUS REQ. 68K has to access the Z-80 S-RAM by byte.

III FM SOUND CONTROL

① 68K accesses the FM source.

68K needs BUS REQ when accessing the FM source, becouse this memory is controlled by Z-80.

② Z80 accesses the FM source.

Z-80 normally controls the FM (4000H - 4003H)

IV PSG CONTROL

PSG accepts access of 68K and Z-80 anytime, but you have to coordinate 68K and Z-80 accesses.
PSG is in $C00011 from 68K and in 7F11H from Z-80.

rcv'd 2/6/90

CONFIDENTIAL

Lisa Chings

copy

OVERVIEW

The Yamaha 2612 Frequency Modulation(FM) sound synthesis IC resembles the Yamaha 2151(used in Sega's coin-op machines) and the chips used in Yamaha's synthesizers.

It's capabilities include:

-- 6 channels of FM sound
-- An 8-bit Digitized Audio channel(as replacement for one of the FM channels)
-- Stereo output capability
-- One LFO(low freqency oscillator) to distort the FM sounds
-- 2 timers, for use by software

To define these terms more carefully; an FM channel is capable of expressing, with a high degree of realism, a single note in almost any instrument's voice. Chords are generally created by using multiple FM channels.

The standard FM channels each have a single overall frequency and data for how to turn this frequency into the complex final waveform(the voice). This conversion process uses four dedicated channel components called "operators", each possessing a frequency(a variant of the overall frequency), an envelope, and the capability to modulate its input using the frequency and envelope. The operator frequencies are offsets of integral multiples of the overall frequency.

There are two sets of three FM channels, named channels 1 to 3 and 4 to 6 respectively. Channels 3 and 6, the last in each set, have the capability to use a totally separate frequency for each operator rather than offsets of integral multiples. This works well(I believe) for percussion instruments, which have harmonics at odd multiples such as 1.4 or 1.7 of the fundamental.

The 8-bit Digitized Audio exists as a replacement of FM channel 6, meaning that turning on the DAC turns off FM channel 6. Unfortunately, all timing must be done by software -- meaning that unless the software has been very cleverly constructed, it is impossible to use any of the FM channels at the same time as the DAC.

2608.2
ADPCM

Stereo output capability means that any of the sounds, FM or DAC, may be directed to the left, the right, or both outputs. The stereo is output only through the headphone jack.

The LFO, or Low Frequency Oscillator, allows for amplitude and/or frequency distortions of the FM sounds. Each channel elects the degree to which it will be distorted by the LFO, if at all. This could be used, for example, in a guitar solo.

Finally, the system has two software timers, which may be used as an alternative to the Z80 VBLANK interrupt. Unfortunately, these timers do not cause interrupts -- they must be read by the software to determine if they have finished counting.

A LITTLE BIT ABOUT OPERATORS

There are four dedicated operators assigned to every channel, with the following properties:

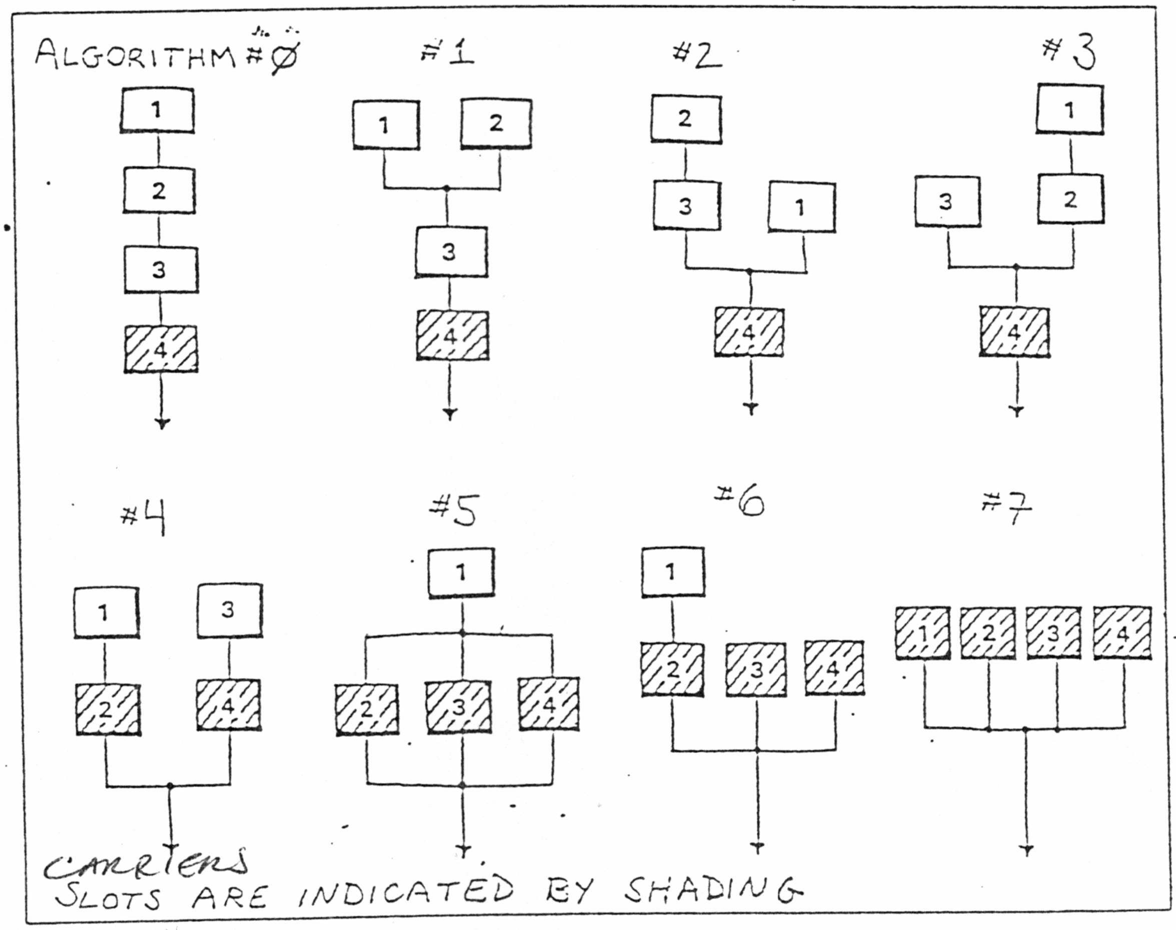
-- An operator has an input, a frequency and envelope with which to modify the input, and an output.

-- The operators have two types, those whose outputs feed into another operator, and those that are summed to form the final waveform. The latter are called "slots".

-- The slots may be independently enabled, though Sega's software always enables or disables them all simultaneously.

-- Operator 1 may feed back into itself, resulting in a more complex waveform.

These operators may be arranged in eight different configurations, called "algorithms". A diagram of the algorithms follows on the next page.



Algorithm 0 -- distortion guitar, "high hat chopper"(?) bass
Algorithm 1 -- harp, PSG(programmable sound generator) sound
Algorithm 2 -- bass, electric guitar, brass, piano, woods
Algorithm 3 -- strings, folk guitar, chimes
Algorithm 4 -- flute, bells, chorus, bass drum, snare drum, tom-tom
Algorithm 5 -- brass, organ
Algorithm 6 -- xylophone, tom-tom, organ, vibraphone
-- snare drum, base drum
Algorithm 7 -- pipe organ

REGISTER OVERVIEW

The system is controlled by means of a large number of registers. General system registers are:

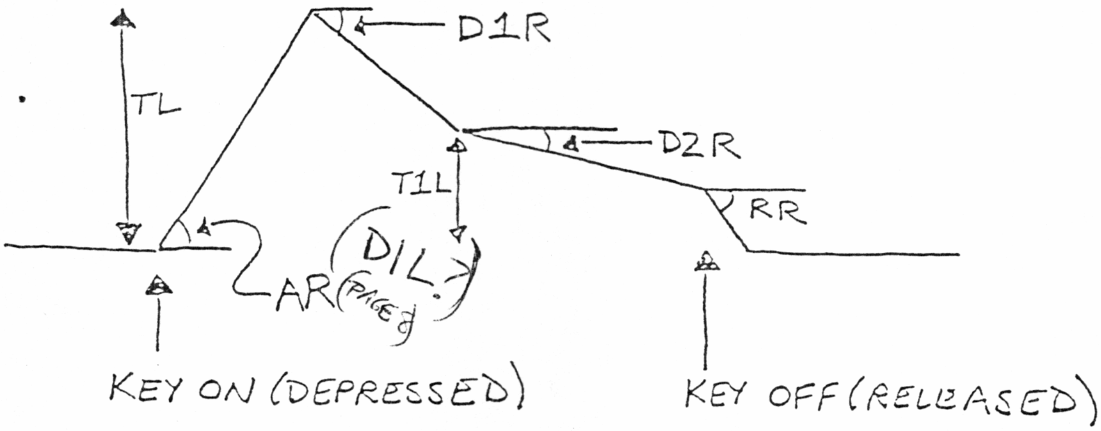
-- timer values and status, software use
-- LFO enable and frequency, to distort the FM channels
-- DAC enable and amplitude
-- output enables for each of the 6 FM channels
-- number of frequencies to be used in FM channels 3 and 6
Usually, an FM channel has only one overall frequency, but if so elected, FM channels 3 and 6 use four separate frequencies, one for each operator.

The remainder of the registers apply to a single FM channel, or to an operator in that channel. Registers that refer to the channel as a whole are:

-- frequency number(in the standard case)
-- algorithm number
-- extent of self-feedback in operator 1
-- output type, to L, R, or both speakers. This can only be heard if headphones are used. (default both)
-- the extent to which the channel is distorted by the LFO. (default none)

Registers that refer to each operator make up the remainder. The four operator's connections are determined by the algorithm used, but the envelope is always specified individually for each operator. In the case of FM channels 3 and 6, the frequency may be specified individually for each operator.

ENVELOPE SPECIFICATION



The sound starts when the key is depressed, a process called "key on". The sound has an attack, a strong primary decay, followed by a slow secondary decay. The sound continues this secondary decay until the key is released, a process called "key off". The sound then begins a rapid final decay, representing for example a piano note after the key has been released and the damper has come down on the strings.

The envelope is represented by the above amplitudes and angles, and a few supplementary registers. Used in the above diagram are:

TL -- Total level, the highest amplitude of the waveform
AR -- Attack rate, the angle of initial amplitude increase. This can be made very steep if desired. The problem with slow attack rates is that if the notes are short, the release(called "key off") occurs before the note has reached a reasonable level.
D1R -- The angle of initial amplitude decrease
T1L -- The amplitude at which the slower amplitude decrease starts
D2R -- The angle of secondary amplitude decrease. This will continue indefinitely unless "key off" occurs.
RR -- The final angle of amplitude decrease, after "key off".

Additional registers are:

RS -- Rate scaling, the degree to which envelopes become shorter as frequencies become higher. For example, high notes on a piano fade much more quickly than low notes.

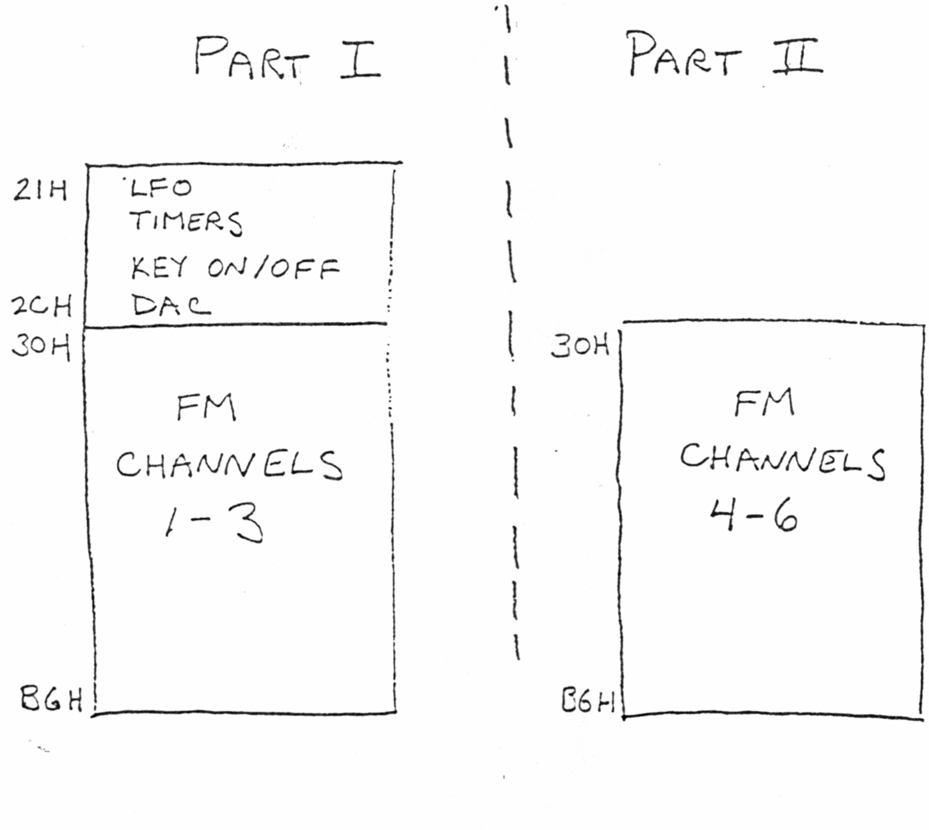
AM -- Amplitude Modulation enable, whether or not this operator will allow itself to be modified by the LFO. Changing the amplitude of the slots(those colored gray in the diagram on page 3) changes the loudness of the note; changing the amplitude of the other operators changes its flavor.

SSG-EG -- a proprietary register whose usage is unknown. It should be set to 0.

The FM-2612 may be accessed from either the 68000 or the Z-80. In both cases, however, the bus is only 8-bits wide.

The FM-2612 is accessed through memory locations 4000H ~ 4003H in the Z80 case, or A04000H ~ A04003H in the 68000 case. These will be referred to as 4000 to 4003.

The internal registers of the FM-2612 are divided as follows;



To write to PART I, write the 8-bit address to 4000 and the data to 4001. To write to PART II, write the 8-bit address to 4002 and the data to 4003.

CAUTION: before writing read from any address to determe if the YM-2612 I/O is still busy from the last write. Delay until bit 7 returns to Ø.

CAUTION: in the case of registers that are "ganged together" to form a longer number — for example the 10-bit Timer A value or the 14-bit frequencies, write the high register first.

READ DATA: reading from any of the four locations

D7							DØ
BUSY	X	X	X	X	X	OVERFLOW B	A

BUSY – 1 if busy, Ø if ready for new data
OVERFLOW – 1 if the timer has counted up and overflowed. See register 27H.

PART I MEMORY MAP

22H	X	X	X	X	LFO EN	LFO FREQ		
24H	TIMER A							
25H	X	X	X	X	X	X	TIMER A	
26H	TIMER B							
27H	CH 3 MODE	RESET B	RESET A	ENABLE B	ENABLE A	LOAD B	LOAD A	
28H	OPERATOR	X	CHANNEL					
2AH	DAC							
2BH	DAC EN	X	X	X	X	X	X	X

30H+	X	DT1	MUL		
40H+	X	TL			
50H+	RS	X	AR		
60H+	AM	X	X	D1R	
70H+	X	X	X	D2R	
80H+	D1L (OR TIL)	RR			
90H+	X	X	X	X	SSG-EG

30H	CH 1, OP 1
31H	CH 2 "
32H	CH 3
	//////////
34H	CH 1, OP 2
35H	CH 2 "
36H	CH 3
	//////////
38H	CH 1, OP 3
39H	CH 2 "
3AH	CH 3
	//////////
3CH	CH 1, OP 4
3DH	CH 2 "
3EH	CH 3
	//////////

Each of 30H-90H has twelve entries, three channels × four operators.
Channels 1-3 become channels 4-6 in PART II.

PART I MEMORY MAP (CONT.)

A0H+	FREQ. NUM				
A4H+	X	X	BLOCK	FREQ. NUM	
A8H+	CH 3 SUPPLEMENTARY FREQ. NUM				
ACH+	X	X	CH 3 SUPP BLOCK	CH 3 SUPP FREQ NUM	
B0H+	X	X	FEEDBACK	ALGORITHM	
B4H+	L	R	AMS	X	FMS
			2 bits		3 BIT

Each of the above has three entries. All follow the pattern

A0H	CH 1 . .
A1H	CH 2
A2H	CH 3
	/ / / / / / / /

with the exception that A8H and ACH follow the pattern

A8H	CH 3, OP 2
A9H	CH 3, OP 3
AAH	CH 3, OP 4
	/ / / / / / / /

"PART II" is a duplication of 30H - B4H, where channels 1-3 are replaced by 4-6.

THE REGISTERS:

22H	X	X	X	X	LFO EN	LFO FREQ

LFO EN - 1 is enabled, Ø disabled

LFO FREQ

	0	1	2	3	4	5	6	7
Hz	3.98	5.56	6.02	6.37	6.88	9.63	48.1	72.2

The LFO (Low Frequency Oscillator) is used to distort the FM sounds' amplitude and phase. It is triply enabled, as there is

a) a global enable in reg. 22H

b) a sensitivity enable on a channel by channel basis, in regs B4H-B6H

c) an amplitude enable on an operator by operator basis, in regs. 60H - 6EH

If the LFO is desired, enable it by register 22H. Next, select which channels will be affected by the LFO, to what degree, and whether their amplitude or frequency is

affected, by setting registers B4-B6H. Finally, if a channel's amplitude is affected, make sure that it is only the "slots" that are affected by setting registers 60H - 6EH.

5k
A = 1024 - 250/18
= 1013
= 3F5

24H	TIMER A MSBs						
25H	X	X	X	X	X	X	TIMER A LSBs

16.6 ms
18 * (1028 - A) = 16666
1024 - A = 16666/18
A = -16666/18 + 1024
= 99
= 63h

Registers 24H and 25H are ganged together to form 10-bit TIMER A, with register 25H containing the least significant bits. They should be set in the order 24H, 25H. The timer lasts

4 ms
18 * (1024 - A) = 4000
1024 - A = 4000/18
A = -4000/18 + 1024
= 801
= 321h

100 µs
18 * (1024 - A) = 100
1024 - A = 100/18
A = -100/18
= 1015
= 3F[illegible]

$$18 * (1024 - \text{TIMER A}) \text{ microseconds}$$

TIMER A = all 1's ⇒ 18 µs = 0.018 ms

TIMER A = all 0's ⇒ 18,400 µs = 18.4 ms

26H | *TIMER B |
|---|

8-bit TIMER B lasts

288 * (256 - TIMER B) microseconds

TIMER B = all 1's ⇒ 0.288 ms

TIMER B = all 0's ⇒ 73.44 ms

$288 * (256 - B) = 4000$

$256 - B = \frac{4000}{288}$

$B = -\frac{4000}{288} + 256$

$= F2$

27H

CH 3 MODE	RESET B	RESET A	ENABLE B	ENABLE A	LOAD B	LOAD A

Register 27H controls the software timers and the Channel 3 (and 6) mode, two entirely separate items.

CH 3 MODE	D7	D6	
NORMAL	0	0	Channel 3 is the same as the others
SPECIAL	0	1	Channel 3 has 4 separate frequencies
ILLEGAL	1	X	—

a normal channel's operators use offsets of integral multiples of a single frequency. In special mode, each operator has an entirely separate frequency. Channel 3 operator 1's frequency is in regs. A2 and A6. Operators 2 to 4 are in regs A8 and AC, A9 and AD, and AA and AE respectively.

No one at Sega has used the timer feature, but the Japanese manual says;

LOAD - 1 starts the timer, 0 stops it.

ENABLE - 1 causes timer overflow to set the read register flag. 0 means the timer keeps cycling without setting the flag.

RESET - writing a 1 clears the read register flag, writing a 0 has no effect.

28H	OPERATOR	X	CHANNEL

This register is used for "Key on" and "Key off". "Key on" is the depression of the synthesizer key, "Key off" is its release. The sequence of operations is; set parameters, key on, wait, key off. When key off occurs, the FM channel stops its slow decline and starts the rapid decline specified by "RR", the release rate.

In a single write to register 28 H, one sets the status of all operators for a single channel. Sega always sets them to the same value, on (1) or off (Ø). Using a special Channel 3, I believe it is possible to have each operator be a separate note, so there is possible justification for turning them on and off separately.

OPERATOR 4 | 3 | 2 | 1	X	CHANNEL

D2	D1	DØ	
0	0	0	Channel 1
0	0	1	2
0	1	0	3
1	0	0	Channel 4
1	0	1	5
1	1	0	6

2AH

DAC DATA

Register 2AH contains 8 bit DAC data.

2BH

DAC EN	X	X	X	X	X	X	X

If the DAC enable is 1, the DAC data is output as a replacement for channel 6. The only channel 6 register that affects the DAC is the stereo output portion of reg B4H.

Registers 30H - 90H are all single-operator registers. Please see page 8 for how the twelve channel-operator combinations are arranged

30H+	X	DT1	MUL

Both DT1 (Detune) and MUL (multiple) relate the operator's frequency to the overall frequency.

MUL ranges from 0 to 15_{10}, and multiplies the overall frequency, with the exception that 0 results in multiplication by ½. That is, MUL=0 to 15 gives ×½, ×1, ×2, ... ×15.

DT1 gives small variations from the overall frequency × MUL. The MSB of DT1 is a primitive sign bit, and the two LSBs are magnitude bits. See the next page for a diagram

D6	D5	D4	MULTIPLICATIVE EFFECT
0	0	0	No change
0	0	1	$\times(1+\epsilon)$
0	1	0	$\times(1+2\epsilon)$
0	1	1	$\times(1+3\epsilon)$
1	0	0	No change
1	0	1	$\times(1-\epsilon)$
1	1	0	$\times(1-2\epsilon)$
1	1	1	$\times(1-3\epsilon)$

where ϵ is a small number.

40H+	X	TL

TL (total level) represents the envelope's highest amplitude, with 0 being the largest and 127_{10} the smallest. A change of one unit is about 0.75 dB.

To make a note softer, only change the TL of the slots (the output operators). Changing the other operators will affect the flavor of the note.

50H+	RS	x	AR

Register 50H contains RS (rate scaling) and AR (attack rate). AR is the steepness of the initial amplitude rise, shown on page 4.

RS affects AR, D1R, D2R and RR in the same way. RS is the degree to which the envelope becomes narrower as the frequency becomes higher.

The frequency's top five bits (3 octave bits and 2 note bits) are called KC (key code) in the following rate formulas:

RS=0 => Final Rate = 2 * Rate + (KC/8)
RS=1 => " " (KC/4)
RS=2 => " " (KC/2)
RS=3 => " " KC

always rounded down.

As rate ranges from 0-31, this means that the RS influence ranges from small (at 0-3) to very large (at 0-31)

60H+	AM	X X	D1R

. D1R (first decay rate) is the initial steep amplitude decay rate (see page 4). It is, like all rates, 0-31 in value and affected by RS.

AM is the amplitude modulation enable, whether or not <u>this</u> operator will be subject to amplitude modulation by the LFO. This bit is not relevant unless both the LFO is enabled and reg B4's AMS (amplitude modulation sensitivity) is non-zero.

70H+	X X X	D2R

D2R (secondary decay rate) is the long tailoff of the sound that continues as long as the key is depressed.

80H+	D1L	RR

D1L is the secondary amplitude reached after the first period of rapid decay. It should be multiplied by 8 if one wishes to compare it to TL. Again, as TL, the higher the number, the more attenuated the sound.

RR is the release rate, the final sharp decrease in volume after the key is released. All rates are 5 bit numbers, but there are only four bits available in the register. Thus, for comparison and calculation purposes, these four bits are the MSBs and the LSB is always 1. In other words, double it and add one.

90H+	X X X X	SSG-EG

This register is proprietary and should be set to zero.

The final registers relate mostly to a single channel. Each register is tripled; please see the diagram on page 9.

AOH+	FREQ. NUM		
A4H+	x x	BLOCK	FREQ NUM
A8H+	CH 3 SUPP. FREQ. NUM		
ACH+	x x	CH 3 SUPP BLOCK	CH 3 SUPP FREQ NUM

Channel 1's frequency is in AØ and A4H.
Channel 2's frequency is in A1 and A5H.
Channel 3, if it is in normal mode (please see page 12) is in A2 and A6H.

If channel 3 is in special mode:

Operator 1's frequency is in		A2 and A6H
Operator 2	"	A8 and ACH
Operator 3	"	A9 and ADH
Operator 4	"	AA and AEH

The frequency is a 14-bit number that should be set high byte, low byte (e.g. A4H then A0H). The highest 3 bits, called the "block", give the octave. The next 10 bits give position in the octave, and a possible 12-tone sequence is;

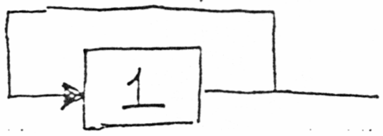
Low	617	C	all in base 10.
	653	C#	
	692	D	
	733	D+	
	777	E	
	823	F	
	872	F+	
	924	G	
	979	G+	
	1037	A	
	1099	A+	
High	1164	B	

This sequence should be used inside each octave.

B0H+

X	X	FEEDBACK	ALGORITHM

FEEDBACK is the degree to which operator 1 feeds back into itself. In the voice library, self feed-back is represented as this:



The ALGORITHM is the type of inter-operator connection used. Please see the list of the eight operators on page 3.

B4H+

L	R	AMS	X	FMS

Register B4H contains stereo output control and LFO sensitivity control.

L - Left output, 1 is on, 0 is off
R - Right output, 1 is on 0 is off

NOTE: the stereo may only be heard by headphones

AMS (amplitude modulation sensitivity) and FMS (frequency modulation sensitivity) are the degree to which the channel is affected by the LFO. If the LFO is disabled, this register need not be set. Additionally, amplitude modulation is also enabled on an operator-by operator level.

AMS	0	1	2	3
dB	0	1.4	5.9	11.8

FMS	0	1	2	3	4	5	6	7
% of a halftone	0	±3.4	±6.7	±10	±14	±20	±40	±80

TEST PROGRAM

Here is a tested power-on initialization and sample note in the "GrandPiano" voice (page 27)

Register	Value	Comments
22H	0	LFO off
27H	0	Channel 3 mode normal
28H	0	all channels off
"	1	
"	2	
"	4	
"	5	
"	6	
2BH	0	DAC off
30H	71H	DT1/MUL
34H	0DH	
38H	33H	
3CH	01H	
40H	23H	Total Level
44H	2DH	
48H	26H	
4CH	00H	

PAGE 26/FINAL PAGE

Register	Value	Comments
50H	5FH	RS/AR
54H	99H	
58H	5FH	
5CH	94H	
60H	5	AM/D1R
64H	5	
68H	5	
6CH	7	
70H	2	D2R
74H	2	
78H	2	
7CH	2	
80H	11 H	D1L/RR
84H	11 H	
88H	11 H	
8CH	A6 H	
90H	0	PROPRIETARY
94H	0	
98H	0	
9CH	0	

Register	Value	Comments
B0H	32H	FEEDBACK/ALGORITHM
B4H	C0 H	Both speakers on
28H	00H	Key off
A4 H	22H	Set frequency
A0H	69H	
28H	F0 H	Key on
<wait>		
28H	00H	Key off

NOTES:

#1 Write address then data

#2 Loop until read register's D7 becomes Ø

#3 Follow MSB/LSB sequence.

PROGRAMMABLE SOUND GENERATOR (PSG)

The PSG contains four sound channels, consisting of three tone generators and a noise generator. Each of the four channels has an independent volume control (attenuator). The PSG is controlled through ~~output port~~ $7F. Mem Loc 7F11

TONE GENERATOR FREQUENCY

The frequency (pitch) of a tone generator is set by a 10-bit value. This value is counted down until it reaches zero, at which time the tone output toggles and the 10-bit value is reloaded into the counter. Thus, higher 10-bit numbers produce lower frequencies.

To load a new frequency value into one of the tone generators, you write a pair of bytes to I/O location $7F according to the following format:

First Byte:	1	R2	R1	R0	d3	d2	d1	d0
Second Byte:	0	0	d9	d8	d7	d6	d5	d4

The R2:R1:R0 field selects the tone channel as follows:

R2	R1	R0	Tone Chan.
0	0	0	#1
0	1	0	#2
1	0	0	#3

10-bit data is: (msb) d9 d8 d7 d6 d5 d4 d3 d2 d1 d0 (lsb)

NOISE GENERATOR CONTROL

The noise generator uses three control bits to select the "character" of the noise sound. A bit called "FB" (Feedback) produces periodic noise or "white" noise:

FB	Noise Type
0	Periodic (like low-frequency tone)
1	White (hiss)

The frequency of the noise is selected by two bits NF1:NF0 according to the following table:

NF1	NF0	Noise Generator Clock Source
0	0	Clock/2 [Higher pitch, "less coarse"]
0	1	Clock/4
1	0	Clock/8 [Lower pitch, "more coarse"]
1	1	Tone Generator #3

NOTE: "Clock" is fixed in frequency. It is a crystal controlled oscillator signal connected to the PSG.

When NF1:NF0 is 11, Tone Generator #3 supplies the noise clock source. This allows the noise to be "swept" in frequency. This effect might be used for a jet engine runup, for example.

To load these noise generator control bits, write the following byte to I/O port $7F:

Out ($7F):	1	1	1	0	0	FB	NF1	NF0

ATTENUATORS

Four attenuators adjust the volume of the three tone generators and the noise channel. Four bits A3:A2:A1:A0 control the attenuation as follows:

A3	A2	A1	A0	Attenuation	
0	0	0	0	0 db	(maximum volume)
0	0	0	1	2 db	NOTE: a higher attenuation results in a quieter sound.
0	0	1	0	4 db	
0	0	1	1	6 db	
0	1	0	0	8 db	
0	1	0	1	10 db	
0	1	1	0	12 db	
0	1	1	1	14 db	
1	0	0	0	16 db	
1	0	0	1	18 db	
1	0	1	0	20 db	
1	0	1	1	22 db	
1	1	0	0	24 db	
1	1	0	1	26 db	
1	1	1	0	28 db	
1	1	1	1	-Off-	

The attenuators are set for the four channels by writing the following bytes to I/O location $7F:

Tone Generator #1:	1	0	0	1	A3	A2	A1	A0
Tone Generator #2:	1	0	1	1	A3	A2	A1	A0
Tone Generator #3:	1	1	0	1	A3	A2	A1	A0
Noise Generator:	1	1	1	1	A3	A2	A1	A0

EXAMPLE

When the Mk3 is powered on, the following code is executed:

```
                LD      HL,CLRTB        ; clear table
                LD      C,PSG_PRT       ; psg port is $7F
                LD      B,4             ; load four bytes
                OTIR                    ; write them
                (etc.)

CLRTB           defb $9F,$BF,$DF,$FF
```

This code turns the four sound channels off. It's a good idea to also execute this code when the PAUSE button is pressed, so that the sound does not stay on continuously for the pause interval.

YM 3438

YM 2612

APPLICATION MANUAL

Yamaha Company, Ltd., Semiconductors Division

[stamped] December 24, 1988 [stamp illegible]

[page 2]

Summary

The Yamaha sound source LSI YM2612 is an FM type LSI sound source with a built-in D/A convertor. The sound source is compatible with OPN (YM2203), and it is capable of generating 6 tones at the same time.

Characteristics

* Simultaneous generation of 6 tones with 4 operators.
* The sine wave has a built-in LFO capability
* Compatible with the YM2203 software.
* A built-in 9 bit D/A convertor. Output for 2 ch.
* Master clock runs at 8 MHz.
* Nch-Si gate MOS LSI.
* Runs on a single 5V power source.
* A 24 pin plastic DIP.

[page 3]

1. Construction and Functions

1-1 Main Functions

The YM 2612 functions are described below. (They are basically identical to OPN (YM2203).

Sound generating mode	.....	4 operators FM type, 6 tones generated simultaneously
Algorithms	.....	8 types
Parameters	.....	see register address and FM sound generating part.

LFO function	.....	Sine wave LFO. Pitch (PM) and amplitude modulation (AM). LFO with variable frequency. PMS and AMS control with a separate on/off AM control for each operator.
Multiple sine wave composition 6 tones.		1 sound can be selected out of
Timer function type B).	.....	two types of timer (type A and
Output control	.....	9 bits with a built-in D/A converter. L and R with an on/off switch.
PCM function	.....	The sampling rate is 55.5 kHz.

[page 4

1 2 Block Diagram

[see the block diagram on page 4]

[page 5]

1 - 3 Terminal Layout Diagram

[see the layout diagram on page 5]

[page 6]

1 - 4

Functions of Terminals

n

This is the input for the master clock.

MOL - MOR

This is an analog output for 2 channels. Its output is provided by the source follower.

D0 ˜ D7

This is an 8 bit bidirectional databus. It works as a processor and it is used to exchange data.

CS - RD - WR - A1 - A0

This serves as a databus controller of D0 ˜ D7.

[table]

1. Address range
2. Content
3. This is used to write the register address of the timer and for similar purposes.
4. This is used to write the register address of channels 1 - 3.
5. This is used to write the register data of the timer and for similar purposes.
6. This is used to write the register data of channels 1 - 3.
7. This is used to writ the register address of channels 4 - 6.
8. This is used to write the register data of channels 4 - 6.
9. This is used to write the status.
10. Changes D0 ˜ D7 to high impedance.

IRQ

This is an interrupt signal generated by the output from 2 timers. When the time period for which the timer is programmed is exceeded, low level is assumed. Output is provided by open drain.

IC

Initializes internal registers.

TEST

This terminal is used to test the main LSI. Please do not connect it to anything.

GND - AGND

This is a grounding terminal.

Vcc - AVcc

This is a power source terminal for + 5 V.

[page 7]

1 - 5 The Databus Controller

The designation of the register address and the control over the data bus to read/write data is implemented by signals CS, WR, RD, and A1. Depending on the status of these signals, description of the content of the controllable addresses range and the content of the data is as displayed on Table 1.1, and an explanation of the functions of the terminals is provided in chapter 1 - 4.

The YM261 register consists of 2 sets of register banks that use jointly addresses $20 - $B6. The bank designation is implemented by the bus controller signal A1. This system makes it possible to access every register.

Register allocation with A 1 is as follows:

When A 1 = "0" When A 1 = "1"

[Figure 1.1]

1. FM common part

Figure 1.1 Allocation of Register Addresses

Table 1.2 explains the allocation of register addresses with CS, WR, RD, AO, and A1 in address control mode.

Table 1.2 Read/Write Selection

[Table 1.2]

1. Mode
2. Address write mode
3. Data write mode
4. Address read mode
5. Inactive mode.

A 1 = "*" designates the bank.

[page 8]

1) Address Write Mode

First of all, the register address is designated in order to write register data from CPU or to read the data from the registers, and after the designation the data is written an it can be read. Address designation when the bus controller signal <address write mode> is written with address data from D0 - D7.

Addresses that have been designated are retained until the next address is designated. Therefore if the same address is accesses continuously, the first writing of the address is valid only once, after that it is invalid.

2) Data Write Mode

After the address has been specified, the bus control signal is changed to <data write mode>, and the data is written to the data bus.

[figure]

1. Address
2. Data
3. Address
4. Data
5. Address
6. Data
7. Data
8. Data
9. W1: Time of retention after address write
10. W2: Time of retention after data write

It is necessary to pay careful attention to the data write process and to the address write process. One of the requirements is a waiting period before switching to the next stage after the writing is finished. This is caused by the method used to process the data inside the LSI.

Consequently, you must set the waiting period as required in order to input the data correctly to the register.

Table 1.3 and Table 1.4 show the waiting periods required during writing to the register.

3) Status Read Mode

Status information which is generated by the status register during the <status read mode> provides output of the bus controller signal to the data bus.

4) Inactive Mode

When <u>CS</u> equals "1", the databus D0 ~ D7 switches to high impedance.

[page 9]

<Waiting Period During the Writing Mode>

Table 1.3 After Address Write

Address	Waiting Cycle
\$21 ~ \$B6	17

Table 1.4 After Data Write

Address	Waiting Cycle
\$21 ~ \$8E	83
\$A) ~ \$B6	47

* The cycle number is the cycle number of the master clock ϕ M.

[page 10]

1 - 6 YM2612 Register Map

A. When ADDRESS is A1 = "0"

B. When ADDRESS is A1 = "1"

1. LFO
   Data
   KEY-ON/OFF
   DAC Data

2. FM Parameter
   Channel 1 ~ 3

3. FM Parameter
   Channel 4 ~ 6

[page 11]

1 - 7 Status Flag

[see the figure on page 11]

[page 12]

2. FM Sound Source - Functions

The YM2612 is an LSI sound source with 4 FM type operators that are capable of generating simultaneously 6 tones/6 tone colors. Because it uses a system with 4 operators, the quality of the FM sound is greatly enhanced, and since the source also includes a built-in LFO capability, the capability to generate tone colors has been dramatically improved. And since it is also provided with a full software compatibility with the YM2203, it is also possible to use tone colors that were created with the YM2203.

This chapter explains the construction of the operators and of each block, with special emphasis on the register capabilities of the YM2612.

2.1 Construction of the Registers

Parameters controlling the operators are located in \$21 ˜ B6. Control over the creation of FM sound source and sound generation is implemented by writing appropriate data to each register.

2 - 1 - 1 Common Registers: \$21 ˜ \$28

Functions common for all the channels of the sound output are located in this block. It contains LFO, a timer, channel allocation capabilities, and other components.

* Test: \$21

This is the test register for YM2612. It is not used for user applications.

* LFO: \$22 (See also 2 - 5 - LFO)

This sets the oscillating frequency determining the speed of the LFO and the on/off control of the LFO.

[page 13]

* Timer - A: \$24, \$25

The timer is a presettable timer containing a timer controller that carries out all timer starting and ending operations as well as the flag control. It also sends the timer interrupt operations to the CPU when IRQ is generated, when at the same time the timer flag allocated to D1 and D0 of the status (read mode) is set to "1".

The timer A has a discrimination capability of 9 μS (M = 8 MHz) of the timer counter, created with 10 bits of \$24 and \$25. Formula {1} explains setting possibilities of timer intervals.

[see figure]

tA = 72 * (1024 - NA) / M .... (1)

NA : 0 ˜ 1023
M : master clock

(Example)

when $M = 8$ MHz

tA (MAX) = 9216 μS
tA (MIN) = 9 μS

* Timer - B : $26

The timer B has a discrimination capability of 144 μS (M = 8 MHz) of the timer counter, created with 8 bits of $26. Formula {2} explains setting possibilities of timer intervals.

[see figure]

$$tB = 1152 * (256 - NB) / M \quad .... (2)$$

NB : 0 ~ 255
M : master clock

[page 14]

* Timer Control: $27

Control is carried out by $27 of the timer controller of the timer A and B.

[figure]

* LOAD: Controls the start and stop operations of the timer.

Setting "1" means start.
Setting "0" means timer stop.

* ENABLE: Controls the timer clock of the status (read mode).

When the setting is "1", the timer counter has an overflow and at the same time "1" is generated on the timer flag. This timer flag also generates the interrupt signal on the IRQ terminal.

When the setting is "0", the flag will not be changed even if the timer counter has an overflow.

* RESET: Resets the timer flag.

When the setting is "1", the timer flag of the status is

reset, and at the same the bit itself is cleared and the setting is changed to "0".

* MODE: Sets the mode of channel 3.

Channel 3 can set the mode by $ 27, "D7" and "D6".

D7	D6	Mode	Function
0	0	normal	Normal sound generation, same as other CH.
0	1	CSM	Generates composite CSM sound mode, F-Number can be set for 4 separate slots. The key on/off uses timer A during the CSM mode.
1	0	effect sound	Same as CSM. It is possible to set an F-Number separately for each slot.

[page 15]

* Key on/off: $28

Keys are assigned for each channel, depending on which channel is specified and whether the slot is on or off.

[figure]

1.	D4	ON/OFF of slot 1
2.	D5	ON/OFF of slot 2
3.	D6	ON/OFF of slot 3
4.	D7	ON/OFF of slot 4

5.	0	0	0	Channel 1
6.	0	0	1	Channel 2
7.	0	1	0	Channel 3
8.	1	0	0	Channel 4
9.	1	0	1	Channel 5
10	1	1	0	Channel 6

*   Test: $2C

This is the test register for YM2612. It is not used for user applications.

2 - 1 - 2 D/A Registers: $2A and $2B

The YM2612 can feed data written from the CPU directly to the D/A converter instead of sending it to the FM sound of channels 6. The register can utilize this function.

*   DAC Data: $2A

Provides data planned for D/A conversion. This data should be prepared as off-set binary data.

Bit 0 is LSB and bit 7 is MSB.

*   DAC Select: $2B

This register is used either to output FM sound to channel 6 or to select the output of data that was written to register $2A.

When bit 7 of this register was set to "0", DAC Data is output when FM sound was set to "1".

[page 16]

2 - 1 - 3 Channel Registers $30 - $B6

This register holds the tone color parameters of 3 channels, F-Numbers, and other components. $30 - $9E are parameters set for the slot units, and $A0 - $B6 are parameters set for the channel units.

The channel selection is controlled by the bus controller signal: A1.

When A 1 = "0", parameter control of CH 1 ~ CH 3.
When A 1 = "1", parameter control of CH 4 - CH 6.

Table 2.1 explains the relationship between register addresses and channel slots.

Table 2. 1 The Relationship between Register Addresses And Channel Slots

1. Parameters

*1: $A8 ˜ $AA and $AC ˜ $AE are set registers for the effect sound mode of the channel 3, and for the frequency (Block, F-Number) and for the composite CSM sound mode. They are not used during the normal sound generating mode.

See also 2 - 1 - 1 for details on how to set channel 3 mode.

[page 17]

2 - 2 Operators

2 - 2 - 1 Operators

The FM sound source can be used to create various tone colors by combining multiple operators. However, 1 operator has only a limited function enabling to conduct calculations based on the data that is written to the registers.

Operators generate the sine wave by the sine generator based on the frequency·information and the output level (envelope).

The block arrangement of the operators is shown on the block diagram below.

[diagram]

1. Frequency information
2. Envelope information

OP: Sine table

PG: Phase generator. This is the frequency (phase) information generating circuit. It sets the speed with which the data of the sine table is read.

EG: Envelope generator. It controls temporary fluctuations of the output level and of the sine table output level.

Specifically, the operator outputs the sine wave when frequency information is furnished to PG and envelope information is furnished to EG. On the other hand, since this method can be used only to obtain the sine wave, the result is not very interesting. For that reason one can also create tone colors that include multiple sound components by connecting multiple operators.

An explanation of the FM principle ... it is possible to create waveforms having composite harmonic sound constructions by modulating the sine wave with the sine wave.

Table 2.1 shows the FM block diagram of 2 operators based on the FM method.

[page 18]

The YM2612 creates FM tone colors by connecting 4 operators. This method of connecting is called the algorithm method, and it is possible to select 8 algorithms.

See also 2 - 2 - 2 for an explanation of algorithms.

An explanation of algorithms ... algorithms describe the mode in which the operators are connected (method of combining operators). In FM there are 8 types of such combinations with 4 operators. The role of each of the operators in creating tone colors is determined by the algorithm, that is to say by the parameters that have been initially set.

Letter β in Figure 2.1 stands for the feedback ratio of the feedback modulation. This feedback depends on the feedback of the input to the identical output, which causes self-modulation. Since this effect is identical with an unlimited number of connected operators, harmonic sound constructions result in integral high

frequencies, which makes it possible to create saw-tooth waves and other appropriate wave forms, for instance strings-like tone colors and other effects. It is possible to think of the feedback capability as if it were one mode. (See also 2 - 2 - 2 for an explanation of algorithms).

The operators are classified as modulators on the modulating side, and they are called carriers on the side with modulated output (irrespectively of whether modulation is present or not). However, the interpretation of the actual function of the operators has not changed, and they can become both modulators and carriers depending on the algorithms.

These names only reflect the fact that when an algorithm has been selected, the role of each operator is either to function as a carrier or to function as a modulator.

On Figure 2.1, OP1 is a modulator and OP2 is a carrier.

The following list of possible initial settings of sound with the FM sound source sums up what was explained above.

A. After selecting an algorithms, the role of each operator is determined.

B. Parameters of the PG (phase generator) are set, and the output frequency of each operator is determined.

C. The parameters of the EG (envelope generator) are set, and the envelope and output level of each operator is determined.

D. Feedback FM is created with tone colors.

[figure]

1. Frequency information
2. Envelope information
3. Frequency information
4. Envelope information
5. β: self-feedback

[page 19]

2 - 2 - 2 Algorithms

Combinations of operators (connection modes) are called algorithms. With an FM method that uses 4 operators, such as the YM2612, it is possible to prepare 4 algorithms, and by selecting the algorithms, the operators can function as carriers or as modulators.

However, the function of the fourth slot is not related to algorithms, and it must be always set as a carrier.

Selection of the algorithms is the most important element of sound generating with the FM sound source. The basic procedure for sound generating is first of all to begin by selecting the most suitable tone color for our purpose. After that parameters are set for each slot, and the tone color is determined.

Figure 2.2 shows the algorithm mode. The characteristics of individual algorithms are explained below.

* Characteristics of Individual Algorithms

1) Serial 4 Set Mode

4 slots are connected in series and multiple modulation is achieved. Because the multiple modulation method is used, a very complex multiple sound construction is achieved with the output of the final carrier as a result of repeated and continuous modulation.

S4 and S3 create the basic tone colors, while S2 and S1 are used to modulate multiple sound components and to add subtle tone coloring.

2) Double Modulation Serial 3 Set Mode

S3 is modulated by the composite output of S2 and S1.
1) By the same token, S4 and S3 create the basic tone color, and a detailed sound composition is created by setting the parameters of S2 and S1.

3) Double Modulation Mode (1)

S4 is modulated by 2 series of modulators. The basic tone color is created by S1 and S4, while S2 and D3 adds a natural quality to the tone color.

4) Double Modulation Mode (2)

3) The double modulation mode is quite similar to the tone described under 3), but since no self-feedback is generated by S3, a flute-like sound, suitable for the sound of wind instruments, is created. S2 and S1 create a noise component.

Since 1) ~ 4) represent 1 carrier, they create a single tone color system, while they also resemble the tone color of solo musical instruments with multiple and composite sound components.

5) Serial 2 Set - 2 Parallel Mode

This is an algorithm with 2 serial constructions of 2 operators. This mode is not particularly suitable for many tone colors with multiple components, but since creating the sound is relatively simple, it is possible to create 2 types of the tone color, which is why this mode can be used to develop a sound that will be characterized by a wide range.

[page 20]

6) Common Modulation - 3 Parallel Mode

The common modulator S1 modulates three carriers, namely S2, S3, and S4.

7) Serial 2 Set + 2 Sine Mode

This creates a composite output of 2 sine waves with 2 operators FM.

8) 4 Parallel - Sine Wave Composite Mode

A composite output with 4 sine waves is obtained in this mode. However, with S1 it is possible to create distorted sound when S1 generates a feedback.

The carriers are used with multiple algorithms as a means to determine the sound quality by those parameters of the generated sound that relate to the frequency information of each carrier.

To give an example, when the algorithm is "7", since multiple values of each parameter will be different, it is possible to achieve a coupler effect, resulting in an organ-like sound quality. In addition, it is also possible to set the detune value to shift somewhat the pitch in order to create a so called coarse (or detuned) effect of the sound quality.

[page 21]

1.	0	Serial 4 Set Mode
2.	1	Double Modulation Serial 3 Set Mode
3.	2	Double Modulation Mode (1)
4.	3	Double Modulation Mode (2)

5. 4 Serial 2 Set - 2 Parallel Mode
6. 5 Common Modulation 3 Parallel Mode
7. 6 Serial 2 Set + 2 Sine Mode
8. 7 4 Parallel Sine Composite Mode
9. M: Modulator
10. C: Carrier
11. Figure 2.2 Algorithms

[page 22]

2 - 2 - 3 Feedback

The first slot of each channel contains the self-feedback function. The feedback has a function of self-modulation, since the operator can be used to return to the input of the modulated signal of its own output. This is expressed as the feedback ratio β, and it can be set in 8 steps from 0 - 7.

The feedback is created in an identical mode by serial connection of multiple operators that are set to the same frequency. In addition, this effect creates a multiple sound construction with integral distribution that is similar to the high frequency element, that is to say a high modulation spectrum that resembles a saw-tooth wave, which is suitable to generate the tone colors that are characteristic for string instruments, a deep level of modulation that is suitable for noise components, and for similar purposes.

[figure]

* FB/Algorithm: $B0 - ~ B2

This is a register that sets the degree of modulation for self-feedback and the algorithms.

[figure]

Table 2.2 shows the degree of modulation for a feedback.

Figure 2.2 The Degree of Modulation for Self-Feedback

1. Degree of modulation

[page 23]

2 - 3 PG (Phase Generator)

The output frequency of the operator, (the speed with which

the furnished phase is read according to the OP sine table), is determined by the frequency (phase) information generated by the PG. In other words, the operator can generate sound by any output frequency by increasing or decreasing the phase value.

2 - 3 - 1 The F-Number and Block

The tones of the musical scale consist of the tones of one octave together with a combination of octaves. Therefore the tones within 1 octave are created by the F-Number and when the Block is used to set the octave information, it is possible to create in a simple manner the musical scale of an 8-octave.

The value of the F-number within 1 octave, which is determined by the master clock and how the frequency required for the sound level is set, can be calculated according to the following formula below.

$$\text{F-Number} = (144 \times f\ \text{note} \times 2^{22} / \quad M) / 2^{\cancel{8}B-1}$$

where F note: the frequency of the generated sound
ϕ M: the master clock
B: block data

(Example)

When M = 8 MHz, A4 (440 Hz) F-Number is required.

$$\text{F-Number (A4)} = (144 \times 440 \times 2^{20} / 8 \times 10^{6}) / 2^{4-1}$$
$$= 1038.1$$

* F Number/Block: \$A0 ˜ \$A2 / \$A4 ˜ \$A6

This is the register where the F-Number and the block data is set. It contains the total of the F-Number for lower 8 bits / upper 3 bits, which is 11 bits, and 3 bits of the Block. This data function as data that is used jointly by 4 operators within the channel.

[figure]

Setting of the F-Number/Block data must be always done according to the following procedure:

(1) Block/F - Num2 data write: \$A4 ˜ \$A6
(2) F-Num1 data write: \$A0 ˜ A2

[page 24]

2 - 3 - 2 Examples of Setting the F-Number Table

ϕ M = 8 MHz, octave: 4 (c4$^{\#}$ ~ C5), A4 = 440 Hz.

[figure]

1. Note
2. Sound level (Hz)
3. F-Number
4. Division

2 - 3 - 3 Creating Key Codes with the F-Number

The Key-Scale function is one of the functions of the envelope generator parameters. The Key-Scale makes it possible to provide scaling that modifies the envelope rate (time) to correspond to the sound level at which the sound is generated.
(See also EG in 2 - 4).

Divisions within 1 octave that are required during key scaling use the F-Number value, and this division data is called the Key Code.

There are 4 divisions in 1 octave which are counted from the upper 4 bits (F11 ~ F8) of the F-Number data.

* Frequency divisions (N4 and N5) within 1 octave.

$$N\ 4 = F11$$
$$N\ 3 = F\ 11 \cdot (F10 + F9 + F8) + \underline{F11} \cdot F10 \cdot F9 \cdot F8$$

In addition, the Block Data consists of 3 bits, and in the entire tone range of 8 octave the Key Code consist of 32 steps.

The following is an explanation of the concept called Detune, which employs frequency divisions using this Key Code.

[page 25]

2 - 3 - 4 Multiple

The term multiple represents a parameter setting the multiplying factor of the frequency information formed by the F-Number. Table 2.3 shows the setting possibilities of the multiplying factor.

* Detune

Detune is a parameter that assigns the frequency information formed by the F-Number as well as a subtle shift of the frequency to each slot unit. In addition to that, Detune is also the value corresponding to the information about every frequency according to the Key Code obtained from the F-Number.

[figure]

Table 2.3 Multiplication Factor with Multiple

1. Multiplication factor

Table 2.4 Detune

1. * D6 is the sine bit.

[page 26]

2 - 4 EG (Envelope Generator)

EG is a circuit generating the volume of sound, which includes both increasing and decreasing the sound volume, as well as generating momentary changes of the tone color.

EG contains an output control circuit which determines the level of the envelope generator and generates the effect of the envelope generator. Envelope information which is required to activate EG is set by each operator through EG parameters that are located in the registers.

2 - 4 - 1 Envelope Generator

The envelope generator generates the envelope that forms temporary modifications of the sound. The envelope has four rates, namely attack, decay, sustain, and release, and it is also displayed through the sustain level.

Figure 2.3 shows an envelope waveform.

[figure]

1. Level
2. Time
3. * The envelope waveform is modified during the attack time when it forms an exponential function. In addition, it changes to a straight line during the rate time.

Figure 2.3 Envelope Waveform and Each Parameter

* AR (Attack Rate): $50 $5E

The attack rate is the speed attained from the Key-on position to the maximum level, and the parameter that is used to determine this rate is called AR. The setting is 5 bits and 32 steps, and the bigger the AR the faster is the rising speed. In addition, when "0" is selected, the attack rate is infinitely great, and modulation of envelope can not be achieved so as not to activate EG.

[figure]

1. * See following items for an explanation of KS.

* DR (Decay Rate): $60 ˜ $6E

The decay rate is the speed of the decrease from the maximum level to the sustain level, and the parameter that is used to determine this rate is called DR. The setting is 5 bits and 32 steps, and the bigger the DR the faster is the decrease of the speed. In addition, when "0" is selected, the decay rate is infinitely great and continuous sound on the maximum level is attained.

[figure]

1. * See 2 - 5 LFO for an explanation of AMON.

[page 27]

* SL (Sustain Level); $80 - $8E

The sustain level is the level (decreasing volume) from the decay rate to the point of the changeover of the sustain level, and the parameter that is used to determine this level is called SL.

The setting is 4 bits and 16 steps, and the greater the SL, the greater is the amount of the decrease. When "" is selected, the amount of the decrease is "0", and it becomes impossible to obtain the effect of decreasing through decay. Table 2.5 explains the arrangement of the bits.

[figure]

1. * When D7 ~ D4 are all "1" (15), 93 dB will be selected.

* SR (Sustain Rate): $70 ~ $7E

The sustain rate is the speed of the decrease from the sustain level, and the parameter that is used to determine this speed is called SR. The setting is 5 bits and 32 steps, and the greater the SR the faster the decrease. In addition, when "0" is selected, the sustain rate is continuous.

[figure]

* RR (Release Rate): $80 ~ $8E

The release rate is the speed of the decrease after Key-off, and the parameter that is used to determine this rate is called RR. The setting is 4 bits and 16 steps, and the greater the RR the faster the decrease.

The operator envelope is created by setting aforementioned parameters A., D., S., and R. However, through these parameters it is only possible to obtain effects which are not connected with the output frequency of the operator, which is usually supplied to the operator with the same rate, and the tone is sometime unnatural.

That is why each rate is modulated to correspond to the sound level of each rate by using the F-Number / Block data. This function is called key scaling

* KS (Key-Scale): $50 ~ $5E

The key scale is a function that enables modulation through the sound level during the envelope period. In other words, the higher the tone, the shorter becomes each rate. A possible setting is 2 bits and 4 steps, with 0 no effect is obtained, and the time difference is greatest when 3 is selected.

Table 2. 6 shows the key scaling values through KS.

[page 28]

Each rate of the envelope generator is ultimately determined by the data of A., D., S., and R. parameter settings, and by the key-scaling values. The values are shown below.

Rate = @ R + Rks ; when R = 0, Rate = 0

* R is the set value for each parameter of D., S., and R. However, for RR (Release Rate) R is (set value x 2 + 1)

* Rks is the key scaling value.

* As the maximum rate is 63, even if the value of the result of the calculations is greater than 63, the rate is always 63.

Table 2.6 Rate Key Scaling

[Table 2.6]

[page 29]

2 - 4 - 2 SSG-type Envelope Control

The envelope of the envelope generator can be controlled by using a previously reset SSG-type envelope. The SSG-type Envelope has the same waveform as the envelope seen in SSG sound source, and it is possible to add modulation of the envelope that can not be obtained simply by setting the EG parameters. Figure 2.4 explains the envelope modes.

All the EG parameters when this envelope is used are listed below.

(1) The setting of the AR data is should be specified as "$1F".
(2) Modulation of the envelop in the key-on mode is determined by level settings of each rate of DR and SR and by SL.
(3) RR works just like a normal mode, the settings of the decrease period after Key-off is selected.

* SSG - EG : $90 ˜ 9E

[figure]

Table 2.4 SSG-type envelope

[page 30]

2 - 4 - 3

The level of the envelope created with the envelope generator is set by an output control circuit. It determines the output level of the operator, its timing range is 96 dB, and its discrimination can be set to 0.75 dB.

It should be emphasized in this context that the output level is shown by the amount of decrease. In other words, the amount of the decrease is set when we take as the maximum value of the output of the operator at 0 DB.

*   TL (Total Level) : $40 ~ $4E

The setting of the output level is done with the total level. The position of each bit reflected in the amount of decrease. Consequently, "00" is selected with 0dB (level max), and "7F" will be the amount of decrease of 96 dB (level min).

However, since the YM2612 has a built-in D/A converter with 9 bits, the real analog sound output corresponds to 54 dB. Consequently, when we compare this output to OPN, the resulting analog sound output is ultimately sometime created as an overflow or underflow even if totally identical parameters are set.

[figure]

[figure]

Figure 2.8 The Position of Each TL bit

1. The amount of decrease (dB)

[page 31]

2 - 5 LFO: Low Frequency Oscillator

LFO modifies the operators with the output of the built-in low frequency oscillator and thus provides a periodic function with respect to the sound. Since the LFO waveform of the YM2612 is a sine wave, modulation control is done through 5 types of parameters.

*   LFO FREQ.: $22

This sets the oscillation frequency determining the speed of the LFO and the on/off control of the LFO.

[figure]

When D3: "1", LFO is on.
When D2 ~ D0: Setting of the oscillation frequency.

[figure]

* PMS (Phase Modulation Sensitivity): $B4 - $B6

LFO is added to (modulated by) the frequency (phase) information that was set by the F-Number/Block, which is how one can periodically modulate the sound level. PMS is a parameter that sets the depth of the modulation and the degree of phase modulation to the channel units.

* AMS (Amplitude Modulation Sensitivity)

:$B4 ~ B6

The output level of the operator can be periodically modulated by adding LFO to the Total Level. AMS is a parameter that sets the depth of the modulation and the degree of amplitude modulation to the channel units.

The effect of the amplitude modulation on the sound that is caused by the LFO depends on the role of the operators. In other words, when the carrier was modulated, the volume of sound is modified, and the tone color is modified by the modulator.

[figure]

1. The degree of modulation (cent)
2. The degree of modulation (dB)

[page 32]

* AMON: $60 ~ $6E

This is a switch the performs the on/off operation for amplitude modulation for each slot. The switch is on when "1" is selected.

[figure]

The parameters described above are used for setting of the LFO.

2 - 7 - 2 Output Selection

The YM2612 provides analog output of two channels called MOL and MOR, and it can distribute the FM sound of 6 channels, or the FM sound + 1 PCM of 5 channels both to the MOL and the MOR.

* L/R: $B4 ~ $ B6 "D7 and D6"

When "1" is selected, this function is ON and the output is provided to the appropriate channel.

On the other hand, some sound will be heard even if "0" has been specified, depending on the set tone color parameters and the total sound level.

[page 34]

3. Electric Characteristics

3 - 1 Absolute Maximum Rating

Item	Rated Value	Unit
Terminal voltage	- 0.3 ~ 7.0	V
Ambient temperature of operational environment	0 - 70	°C
Storage temperature	- 50 ~ 125	°C

3 - 2 Conditions Recommended for Operation

Item	Symbol	Minimum	Standard	Maximum	Unit
Voltage of the power source	Vcc	4.75	5.0	5.25	V
	GND		0		V

[page 35]

3 - 3 Direct Current Characteristics

(Ambient Temperature of the Operational Environment is $T_a = 0 \sim 70$ °C)

1. Item
2. Symbol
3. Conditions
4. Minimum
5. Standard
6. Maximum
7. Unit
8. Input high level voltage
9. Total input
10. Input low level voltage
11. Total input
12. Clock input high level voltage
13. Clock input low level voltage
14. Input leak current
15. Three state input current (off state)
16. Output high level voltage
17. Output low level voltage
18. Output leak current (off state)
19. Analog output voltage
20. Maximum sound volume
21. Power source current
22. Pull up resistance
23. Input capacity
24. Total input
25. Output capacity
26. Total output

[page 36]

3 - 4 Alternating Current Characteristics

(Ambient Temperature of the Operational Environment is $T_a = 0 \sim 70$ °C)

1. Item
2. Symbol
3. Conditions
4. Minimum
5. Standard
6. Maximum
7. Unit
8. Input clock frequency

9. Input clock duty
10. Input clock rising time
11. Input clock falling time
12. Address setup time
13. Address hold time
14. Chip select write width
15. Write pulse width
16. Write data setup time
17. Write data hold time
18. Chip select read width
19. Read pulse width
20. Read data access time
21. Read data hold time
22. (Figure 3 - 1)
23. (Figure 3 - 1)
24. (Figure 3 - 1)
25. (Figure 3 - 2, 3)
26. (FIgure 3 - 2, 3)
27. (Figure 3 - 2)
28. (Figure 3 - 2)
29. (Figure 3 - 2)
30. (Figure 3 - 2)
31. (Figure 3 - 3)
32. (Figure 3 - 3)
33. (Figure 3 - 3) C_L = 100 pF

34. * Reset

35. Item
36. Symbol
37. Conditions
38. Minimum
39. Standard
40. Maximum
41. Unit
42. Reset pulse width
43. (Figure 3 - 4)
44. Cycle

45. 3 - 5 DAC Characteristics

46. Item
47. Symbol
48. Conditions
49. Minimum
50. Standard
51. Maximum
52. Unit
53. Maximum output amplitude
54. Discrimination

[page 37]

1. The standard of the setting on the timing diagram is VH = 2.0 V, VL = 0.8 V.

2. Figure 3 - 1 Clock timing
3. (Note)
4. Tcsw, Tww, Twds, and Twdh have a high level of CS and WR as a standard.

5. Figure 3 - 2 Write Timing
6. (Note)
7. Tacc have a LOW level of either CS or RD as a standard.
8. Tcsr, Trw, Trdh have a HIGH level of either CS or RD as a standard.
9. Figure 3 - 3 Read Timing

[page 38]

1. Figure 3 - 4 Reset - Pulse Width

SEGA™

SEGA OF AMERICA, INC.
Consumer Products Division

GENESIS TECHNICAL BULLETIN #11

To: Developers and Third Parties

From: Mac Senour, Technical Support MS

Date: 9/9/91

Re: Problems during sound access

Sound output stops during a game.

Problem:

The busy flag was read in the FM sound generator YM2616 like this (address 4001h)

(CS,RD,WR,A1,A0) = (0,0,1,0,1)

However, in the case of the YM2612, it's output is not regulated according to the conditions set above. This results in the device being read as "not busy" and as a consequence, ends up outputing sound.

Fix:

When the FM sound generator's busy flag is read, do not access anything else other than:

(A1,A0) = (0,0) (address 4000h)

130 Shoreline Drive, Redwood City, CA 94065 • (415) 508-2800 • FAX (415) 802-1448

SEGA™

SEGA OF AMERICA, INC.
Consumer Products Division

GENESIS TECHNICAL BULLETIN #12

To: Developers and Third Parties

From: Mac Senour, Technical Support Mac

Date: 9/9/91

Re: Problems with repeated resets

The software seems to go out of control when resets are repeated.

The Problem:

When the reset occurs the CPU is reset, however the VDP is not. When the reset occurs during at DMA the VDP continues the DMA. The VDP is accessed right after the reset. If this is done while the VDP is executing a DMA then this access is ineffective.

The Fix:

Before accessing the VDP after the initialization program (ICD_BLK4), check the DMA BUSY status register. If a DMA is being executed, do not attempt to access the VDP.

If the problem persists, ensure that you are not executing a DMA right after a reset.

130 Shoreline Drive, Redwood City, CA 94065 • (415) 508-2800 • FAX (415) 802-1448

SEGA OF AMERICA, INC.
Consumer Products Division

GENESIS TECHNICAL BULLETIN #13

To: Developers and Third Parties

From: Mac Senour, Technical Support Mac

Date: 9/9/91

Re: Corrections to the Genesis Software Manual

When discussing VRAM, CRAM and VSRAM access, the manual states in pages 22-27 that byte access is possible. This is incorrect. Access is limited to word or long word.

On page 77 it implies that the 68000 may set the bank switches. The bank switches MUST be set by the Z80.

These changes affect version 1.0 of the manual, later versions will reflect this correction.

130 Shoreline Drive, Redwood City, CA 94065 • (415) 508-2800 • FAX (415) 802-1448

SEGA

SEGA OF AMERICA, INC.
Consumer Products Division

GENESIS TECHNICAL BULLETIN #14

To: Developers and Third Parties

From: Mac Senour, Technical Support

Date: 9/5/91

Re: ROM splitting

As we all know Genesis products must be split into 128k odd & even pieces. Sega expects the ROM images to be in the following format:

0:Even	1:Odd
2:Even	3:Odd
.	
.	
.	

For larger products, continue the above pattern. We would appreciate it if all products would conform to this method of splitting. Please request the utility to split ROMs, M2B or Split4, if your current tool can't output files in this manner.

130 Shoreline Drive, Redwood City, CA 94065 • (415) 508-2800 • FAX (415) 802-1448

PC - GENESIS ~~SERIAL~~ CABLE
4 BIT PARALLEL